Doctoral Dissertation

Generative AI-Driven 3D Modeling and Applications using Intuitive Control:

From Architectural Design to Urban Planning

DU Xusheng

Supervisor XIE Haoran

Graduate School of Advanced Science and Technology
Japan Advanced Institute of Science and Technology
(Information Science)

March 2026

Abstract

Global cities and architectures are undergoing profound structural and functional transformations. New buildings must meet higher performance standards under tighter budgets and shorter design cycles, while existing building stocks require large-scale adaptive reuse to avoid wasteful demolition and reduce environmental impact. At the urban scale, historical spatial data and multi-decade development trends are increasingly used to project future urban form, making long-term forecasting indispensable for land allocation and policy formulation. However, the digital tools that should support these tasks have not been sufficiently explored or developed to address these emerging demands. High-fidelity 3D building modeling remains labor-intensive; Level of Detail (LoD) architectural representations are fragmented and inconsistent; facade renovation workflows depend heavily on expert judgment; and city-level prediction tools struggle to integrate density, height, transportation networks, and historical evolution.

Despite the rapid progress of generative Artificial Intelligence (AI), particularly diffusion models and multimodal vision–language models (VLM), these technologies remain fundamentally misaligned with the requirements of the built environment. Most existing models are trained on natural images and therefore optimize primarily for visual plausibility rather than structurally meaningful representation. As a result, they struggle to capture part–whole relationships, spatial organization, and cross-view consistency, which are essential for architectural and urban modeling. Current generative models also lack mechanisms to maintain semantic and geometric continuity across LoD. Sketches, massing models, and detailed facades are treated as unrelated conditions rather than coordinated expressions of the same underlying structure, even though early-stage sketches play a critical role in representing design intent, boundary logic, and spatial hierarchy. Likewise, these models cannot connect sketch intent, component-level geometric logic, and long-term urban evolution into a unified, cross-level reasoning process. Yet effective spatial design requires cross-level continuity between buildings and the cities they constitute, rather than handling them as isolated scales. In practice, existing generative AI methods can generate images that look like buildings, but they cannot yet interpret how buildings are composed, how their components relate, or why urban patterns develop over time. These limitations highlight the need for generative approaches grounded not only in visual appearance but also in structural relationships, hierarchical

representation, and temporal dynamics, which are core attributes shaping how buildings are designed and how cities evolve. More fundamentally, these challenges stem from the absence of a unified cross-level modeling paradigm that can consistently connect object composition, building-scale organization, and city-scale evolution within a generative approach.

To address these challenges, this dissertation argues that generative modeling for the built environment must be formulated as a cross-level problem, and accordingly proposes a unified approach that systematically integrates object-, building-, and city-level spatial representations within a computational paradigm.

- **Object-Level: Model Generation.** As the foundation of the proposed cross-level approach, this part introduces DualShape, a hybrid retrieval–generation framework that combines sketch-guided component retrieval with implicit SDF-based synthesis. By resolving ambiguity and incompleteness in freehand sketches, DualShape enables efficient, topology-consistent 3D component modeling, offering a new pathway for rapid geometric prototyping in early design.
- **Building-Level: Architectural Design and Renovation.** Building upon the object-level understanding of three-dimensional spatial structure and part–whole relationships, this part extends the approach to buildings as a specific class of objects and proposes methods for architectural design and renovation. This part first constructs a multi-view, geometrically aligned Level-of-Detail (LoD) sketch dataset, addressing the long-standing lack of consistent LoD data in architectural research. In addition, a multi-view consistent 3D generation framework is developed, capable of reconstructing building models that remain geometrically aligned across different viewpoints. Furthermore, a facade renovation framework that integrates VLM-based structural reasoning, sketch-conditioned diffusion, and ControlNet refinement provides efficient and structure-aware design support for the adaptive reuse of aging industrial buildings.
- **City-Level: Urban Evolution.** In addition to object- and building-level modeling, the city scale represents a critical level of analysis for the built environment, where long-term spatial dynamics and collective urban patterns emerge. This part proposes MMCN, a Memory-aware Multi-Conditional generation Network that integrates building density, height, transportation networks, and historical urban patterns through multi-ControlNet, semantic fusion, and a spatial memory mechanism. Experiments on Shenzhen's dataset from 2005–2024 demonstrate high predictive performance (SSIM 0.885, Boundary IoU 0.642) and robust

generalization to Shanghai and Tianjin, enabling coherent cross-year and cross-city urban evolution forecasting.

Collectively, these three levels constitute a coherent cross-level generative foundation, within which geometric primitives, architectural semantics, and urban dynamics are modeled under a unified representational logic. This dissertation provides a new computational foundation for interpretable, controllable, and scalable generative design, supporting future applications in architectural practice, adaptive reuse, sustainable urban planning, and intelligent digital twins.

Keywords: Generative AI, Diffusion Models, Sketch-Based Modeling, 3D Generation, Level of Detail (LoD), Architectural Design, Facade Renovation, Urban Evolution, Cross-Level Generation.

Acknowledgment

First and foremost, I would like to express my gratitude to my primary supervisor, Professor Haoran Xie, whose support has been the foundation of my academic growth. His guidance has consistently shaped the direction of my research. Since the beginning of my graduate studies, Professor Xie has provided thoughtful feedback, rigorous academic advice, and encouragement. His mentorship has been instrumental in refining my ideas and methodologies and nurturing my development as an independent researcher.

I am also grateful to my co-supervisor, Professor Shogo Okada, for his guidance and constructive suggestions. His expertise and thoughtful feedback have greatly contributed to the development of this dissertation.

I would also like to express my sincere thanks to Professor Kazunori Miyata, whose profound knowledge and kind guidance have been invaluable to my research.

My gratitude further goes to Professor Shinobu Hasegawa, who supervised my secondary research theme and provided valuable advice and support that broadened my academic perspective.

My sincere appreciation extends to Professor Ye Zhang from Tianjin University, whose professional insight and warm encouragement have played a significant role in advancing my research.

I would also like to thank Professor Ruizhen Hu and Professor Naoya Inoue for serving as members of my dissertation examination committee and for their valuable feedback.

I am also deeply grateful to all co-authors who collaborated with me throughout the projects included in this dissertation, including Chengyuan Li, Ruihan Gui, Yuxiao Ren, Warissara Booranamaitree, and all colleagues who contributed to this work.

I am also grateful for the financial support from JST SPRING, under Japan Grant Number JPMJSP2102, which enabled me to pursue my research.

Finally, I am also thankful to my lab mates and colleagues for their camaraderie, collaboration, and intellectual companionship, which made this experience far more memorable and meaningful.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

This chapter provides the conceptual and methodological context for the dissertation. It first outlines the broader background of generative modeling, architectural computing, and urban analysis, and clarifies why new cross-level approaches are needed. It then presents the practical and theoretical motivations of the study, summarizes the key challenges in architectural design and urban planning, and introduces the proposed methodological framework and research tasks. Finally, it explains how the subsequent chapters are organized to develop and validate the proposed cross-level generative modeling system.

1.1 Background and Significance

The fields of computer vision and Artificial Intelligence (AI) have experienced a fundamental paradigm shift from discriminative perception to generative modeling, transforming AI from systems that recognize and classify the world to those capable of synthesizing and constructing it. Early studies in computer graphics [1] established the theoretical foundation for digital modeling and rendering, enabling designers to reproduce the visual characteristics of the physical world through geometric modeling and photorealistic techniques. With the emergence of deep learning [2–4], computers began to acquire the ability to learn structural and semantic patterns directly from large-scale data, significantly enhancing the accuracy of visual recognition and feature representation. These developments transformed artificial intelligence from systems that merely perceived and described visual inputs into ones capable of understanding complex visual structures.

This evolution has positioned generative AI as a central force in redefining visual and spatial representation. From the introduction of Generative Adversarial Networks (GANs) [5] and the success of Diffusion Models [6, 7] in high-fidelity synthesis, to the emergence of ControlNet [8] and Large Vision–Language Models (VLMs) such as CLIP [9] and GPT-4V [10], these generative methods have evolved from low-level pixel generation toward

Figure 1.1: Models with different levels of detail. LoD1 represents coarse massing, LoD2 provides intermediate structural definition, and LoD3 captures detailed architectural representation.

semantically grounded and structurally coherent spatial creation. Through probabilistic modeling, conditional control, and semantic alignment, these methods enable models to not only synthesize realistic two-dimensional imagery but also infer coherent three-dimensional spatial structures, depth, and geometric relationships. This transition from visual representation to structured spatial generation provides new methodological foundations for three-dimensional modeling and spatial synthesis.

Representation learning and generative modeling have gradually converged within this evolving paradigm. The former uncovers the intrinsic logic of spatial and structural organization through high-dimensional feature abstraction [11, 12], while the latter builds upon it to generate and reconstruct new spatial forms. Their integration drives artificial intelligence from static recognition toward dynamic spatial creation, making data-driven form generation possible and offering new theoretical perspectives for the digital transformation of architecture and urban design.

Meanwhile, architecture itself has been undergoing a profound digital transformation. Since the adoption of Building Information Modeling (BIM), the organization of architectural data has evolved from two-dimensional drawings to multi-layered, semantically enriched three-dimensional models [13]. Parametric and algorithmic design have further advanced the procedural and computational nature of architectural reasoning, enabling designers to explore morphological diversity through parameters, rules, and algorithms [14]. Furthermore, the introduction of Level of Detail (LoD) (Figure 1.1) has made the expression of architectural information more systematic, spanning from simplified massing models to detailed construction models, each corresponding to a specific semantic depth and level of precision [15]. However, despite these advancements in visualization and management efficiency, these

methods largely depend on manual operation and explicit rules, lacking autonomous generative and semantic understanding capabilities.

Importantly, LoD representations inherently encode cross-level relationships in architectural design. They describe not only different degrees of geometric refinement, but also transitions between conceptual abstraction and spatial realization. From early massing studies to detailed architectural components, LoD serves as a representational bridge that links design intent, spatial hierarchy, and constructability. This dissertation builds upon this observation and treats LoD not merely as a visualization standard, but as a core mechanism for enabling cross-level generative modeling.

Generative AI introduces new opportunities for architectural design. By learning from multimodal data such as sketches, layouts, depth maps, and semantic labels, generative models can infer plausible architectural forms and spatial structures from abstract representations, realizing a transition from geometric modeling to semantic generation [16, 17]. This shift transforms the design process from experience-driven to data-driven, and from human–computer collaboration toward model-assisted co-creation. In this context, learning-based generative systems become active cognitive partners capable of understanding design intentions, generating spatial solutions, and assisting creative reasoning, thereby redefining the modes of expression, workflow, and epistemology in architecture.

From a technical perspective, this research aims to establish a Cross-Level Generative Modeling Approach that unifies spatial generation tasks across the architectural and urban levels under a consistent generative logic (Figure. 1.2). The research pursues three interconnected objectives:

1. **Object Level:** Explore the potential of generative models in structured form representation and spatial reconstruction.
2. **Building Level:** Investigate controllable generation mechanisms from sketches and depth maps, enabling models to understand design semantics and LoD hierarchies.
3. **City Level:** Develop a multimodal generative framework integrating layout, density, height, neighborhood context, and temporal sequences to predict urban morphological evolution.

This framework seeks to bridge the traditional gap between architectural and urban research by establishing a continuous generative logic that connects individual-level generation to system-level forecasting, thus providing both theoretical and methodological support for cross-level urban modeling.

From a societal standpoint, this study explores the potential of generative AI in sustainable architecture, building renewal, and intelligent urban planning. With the acceleration of urbanization and the growing demand for

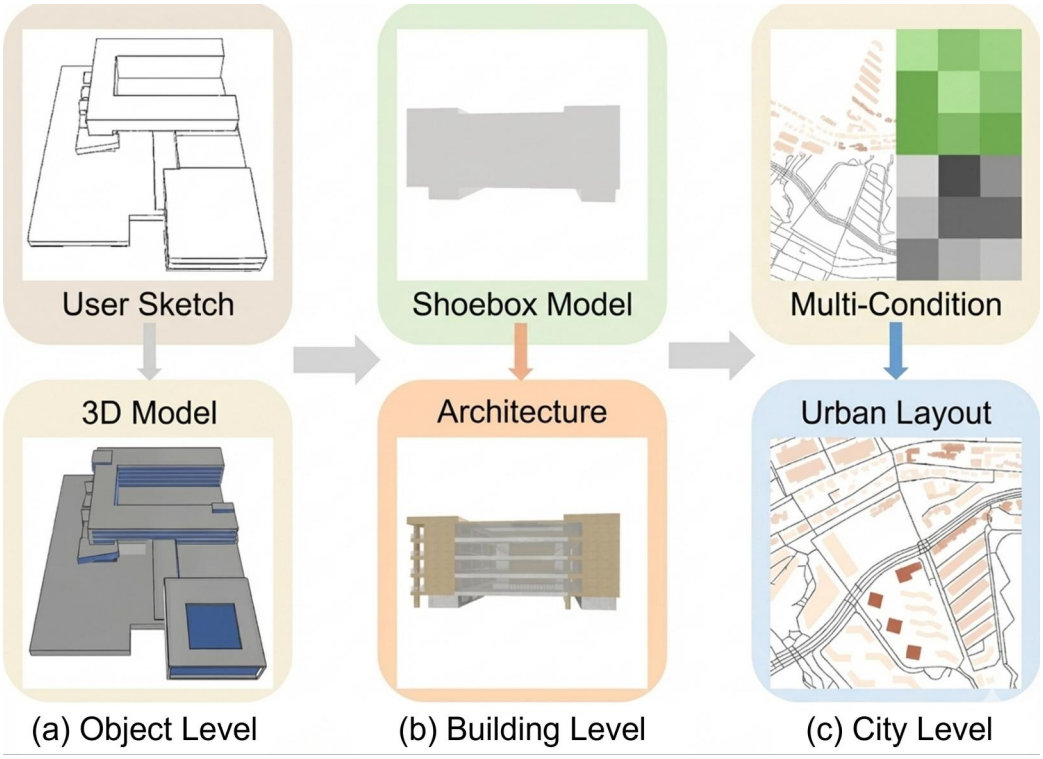


Figure 1.2: Cross-level generative modeling pipeline. It integrates (a) object level: sketch-based 3D generation, (b) building level: architectural LoD modeling, and (c) city level: urban evolution forecasting into a unified multi-level generative workflow.

stock regeneration, the fields of architecture and planning face the challenge of balancing efficiency, innovation, and sustainability [18]. The proposed generative framework, supported by multi-scale data, provides designers and planners with intelligent assistive tools: facilitating early-stage design exploration and enabling large-scale spatial prediction and evaluation. More importantly, this research emphasizes the continuity across 3D representation, architectural design, and urban forecasting. Such a cross-level, integrated generative perspective establishes a new paradigm for AI-driven architecture and urban design, promoting not only technological integration but also intellectual renewal and theoretical expansion within the discipline in the era of computational intelligence [19].

The overarching goal of this research is to evolve artificial intelligence from a mere tool into a genuine co-creator, an intelligent partner capable of understanding architectural semantics, perceiving spatial logic, and forecasting urban evolution. This transformation represents not only technological

advancement but also a conceptual renewal in how design intelligence is defined in the digital era.

1.2 Motivation

The rapid evolution of artificial intelligence and generative modeling has brought architecture and urban design to a transformative juncture. However, when these emerging technologies enter the architectural domain, they encounter a range of complex and practical challenges. Although generative AI has demonstrated remarkable creativity in image, speech, and text synthesis, enabling it to comprehend architectural logic, spatial order, and urban complexity remains an unresolved theoretical and methodological issue. The motivation for this research, therefore, arises not only from technological possibilities but also from the structural constraints and future demands inherent in architectural practice.

1.2.1 Challenges in the Architectural Practice

Despite the proliferation of digital design tools, architectural practice remains largely governed by experience-driven and intuition-based decision-making. In most real-world projects, architects rely heavily on personal expertise and subjective judgment to conceive and iterate design proposals. This process is time-consuming, difficult to quantify, and challenging to reproduce. Whether in conceptual exploration or technical development, decisions are still guided primarily by intuition rather than by systematic data. While this human-centered approach preserves creative freedom, it often leads to inefficiency, lengthy iteration cycles, and the absence of reusable design knowledge.

Information discontinuity across design stages further exacerbates this problem. The spatial intentions embedded in conceptual sketches cannot be seamlessly transformed into computable three-dimensional models, and the data generated during construction rarely informs the early stages of design. Consequently, the design process lacks a closed feedback loop. Historical projects, building models, and urban plans are often fragmented across disparate platforms, without mechanisms for structured integration or reuse. This fragmentation hinders the formation of learnable knowledge bases and limits the development of intelligent design systems.

A more fundamental limitation lies in the persistent gap between architecture and urban planning. Architectural design focuses on the spatial form and functional organization of individual buildings, whereas urban planning operates at a macro scale, addressing system structures and development

strategies. Due to discrepancies in data models, semantic hierarchies, and spatial–temporal resolution, the generative logic linking buildings and cities has not yet been effectively unified. More fundamentally, this reflects a cross-level discontinuity in existing generative approaches, where architectural-scale reasoning and city-scale modeling are developed under separate representations, assumptions, and objectives. As a result, current generative methods in architecture remain confined to localized tasks such as facade generation or morphological completion, without achieving cross-level continuity. This disjunction constitutes a key bottleneck in advancing intelligent generative design.

1.2.2 Emerging Technological Opportunities

Recent advances in generative artificial intelligence offer promising opportunities to overcome these limitations. With the development of Diffusion Models [7] and ControlNet [8], modern generative systems can now generate spatially coherent three-dimensional forms directly from unstructured inputs such as sketches, semantic layouts, density maps, or textual descriptions. This capability transcends the traditional constraints of geometric modeling, allowing designers to communicate intent more intuitively while enabling models to automatically translate abstract concepts into spatially explicit representations.

The rise of multimodal generation further enables generative models to integrate textual, visual, and geometric information, aligning semantic understanding with spatial synthesis. For instance, by conditioning on both text and image, a model can regulate style, proportion, and spatial organization, supporting interpretative and creative design exploration. Such capabilities enhance conceptual efficiency in early-stage design and introduce new modes of interdisciplinary collaboration in complex architectural and urban projects.

Furthermore, the emergence of Large Vision–Language Models (VLMs) is reshaping how architectural knowledge is organized and applied. These models are capable of semantic interpretation and feature retrieval across vast historical datasets, providing a foundation for data-driven design intelligence. By learning from patterns of building distribution, topographic evolution, and functional transformation, generative methods can capture long-term development trends and contribute to predictive urban modeling. In this regard, generative AI functions not merely as a form-generation tool but as a computational framework for knowledge association and spatial reasoning, laying the groundwork for data-driven architectural intelligence.

1.2.3 Early-Stage Design Representations

As generative AI becomes more deeply embedded in architectural and urban workflows, a critical question arises: How can AI engage with the languages through which design intent is expressed? While recent models can generate spatial forms from images, text, or parametric inputs, architectural and urban design typically begin with abstract, schematic representations that externalize conceptual reasoning. Among these, freehand sketches play a particularly important role, especially at the object and building levels of design.

A sketch is more than a graphical notation; it is a visualization of the designer's cognitive process. Within its lines, proportions, and tonal relationships lie preliminary hypotheses about spatial order, massing balance, and structural hierarchy. In early-stage architectural design, sketches serve as both the origin of creative exploration and a key medium through which architectural semantics are formed. Because of their ambiguity and flexibility, sketches enable designers to explore spatial possibilities beyond strict geometric constraints, giving them an irreplaceable position in many architectural practices.

At the same time, sketches are not the only medium through which design intent is communicated. In building-level workflows, floor plans, contour drawings, massing models, and depth or semantic maps provide complementary diagrammatic representations of spatial logic. At the urban level, land-use layouts, density and height maps, road networks, and temporal development diagrams play a similar role, encoding system-level constraints and evolution patterns in a compact, spatially structured form. In this broader sense, sketches can be regarded as one instance within a family of early-stage, schematic representations that bridge human reasoning and computational processing.

For computational models, these schematic media present a common difficulty: they are sparse, abstract, and semantically implicit. Whether in the form of freehand sketches or simplified layouts, they often lack explicit depth cues, structural annotations, or semantic labels, making it challenging for traditional algorithms to infer spatial logic or design intent. As a consequence, existing generative approaches frequently rely on high-resolution imagery or fully annotated geometry, while underutilizing the rich but implicit information carried by early-stage design representations.

Within this dissertation, one methodological line focuses on a "From Sketch to 3D" perspective at the object and building levels, where sketches and derived contours are fused with depth estimation and semantic segmentation to support part-aware 3D generation and multiview architectural

modeling. Another methodological line extends the same principle to non-sketch representations at the urban level, where layout maps, density and height distributions, road networks, and temporal sequences are treated as higher-level schematic inputs for spatial generation and forecasting.

By interpreting sketches and other early-stage representations as machine-readable carriers of design intent, the proposed framework aims to transform them from static records of creativity into dynamic inputs for generative synthesis and spatial inference. In this way, design intentions can be computationally captured, extended, and reused across levels, enabling a continuous transition from conceptual expression to spatial realization. Generative AI thus moves from passively reproducing forms to actively participating in design reasoning, supporting a shift from "humans interpreting drawings" to a collaborative paradigm in which computational models also learn to interpret and respond to the languages of design.

1.2.4 Summary of Research Motivation

In summary, the motivation of this research can be articulated across two dimensions: technical and application-oriented:

- **Technical Motivation:** This study explores how artificial intelligence can understand architecture. Beyond generating visual forms, the objective is to enable AI to learn the semantic logic, hierarchical structure, and spatial relationships embedded in architectural design, achieving what may be termed architectural semantic generation. Through multimodal data fusion and generative mechanism design, the research seeks to advance AI from visual generation to spatial generation.
- **Applicational Motivation:** The study further aims to establish a continuous generative framework that unifies architectural and urban levels, enabling AI to operate coherently across building design, architectural renewal, and urban forecasting. This framework not only offers architects intelligent tools for enhancing creative efficiency and exploration depth but also provides planners with data-driven methods for spatial simulation and evaluation—contributing to sustainable urban development.

From a broader perspective, this research envisions a disciplinary transformation: evolving artificial intelligence from a computational tool into a cognitive collaborator, an intelligent partner capable of understanding sketches, generating architectural forms, and reasoning about urban evolution. This transformation represents not only technological advancement but also a conceptual renewal in how design intelligence is defined in the digital era.

1.3 Challenges

While generative artificial intelligence has opened new possibilities for architecture and urban design, the application of general-purpose generative technologies to the architectural domain still faces significant theoretical and practical limitations. These challenges can be summarized across three levels: the mismatch between technology and architectural semantics, the difficulty of generation and renovation in architectural design, and the complexity and dynamism inherent in urban planning. Identifying and addressing these issues form the direct motivation and core objectives of this research.

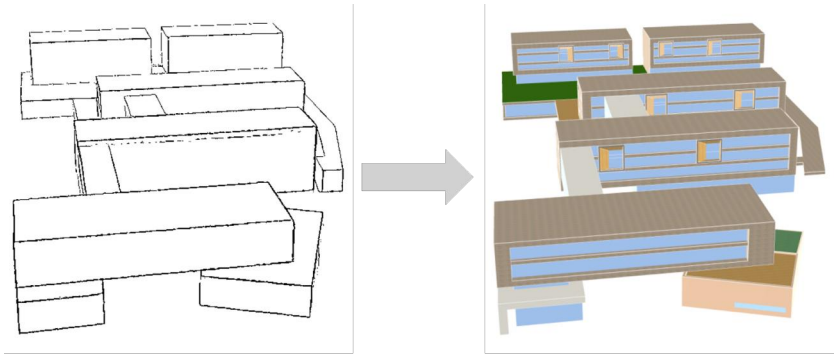
1.3.1 Limitations of Generative Models

Current mainstream generative models, such as Diffusion Models [6, 7] and GANs [5], have achieved remarkable progress in image synthesis and shape generation. However, these models are primarily designed for visual appearance rather than spatial logic. When directly transferred to architectural contexts, they exhibit three key structural limitations.

First, these models lack architectural semantic understanding. Such models tend to learn superficial visual features such as edges, textures, and outlines, without grasping the semantic relationships among architectural elements, such as the hierarchy of volumes, spatial divisions, or structural logic. As a result, the generated buildings may appear visually plausible but often display inconsistencies or discontinuities in spatial function and structural organization.

Second, these methods lack a Level of Detail (LoD) mechanism. Architectural design progresses through multiple stages, including conceptual sketching, mass modeling, facade design, and construction documentation, each corresponding to different levels of geometric precision and semantic richness. Generic generative models, however, typically operate at a single resolution and are unable to represent this hierarchical evolution from abstraction to refinement.

Third, these models struggle to capture spatial logic and neighborhood continuity. Architectural coherence depends not only on the form of individual buildings but also on their interrelationships, scale coordination, and contextual integration. Existing AI models focus largely on isolated object generation and rarely encode the organizational or contextual continuity of architectural spaces. In short, current generative techniques in architecture remain confined to geometric representation and fall short of understanding architectural semantics and spatial logic.



(a) Architectural Generation

(b) Architectural Renovation

Figure 1.3: Challenges in architectural design. (a) Architectural generation: transforming sketches into structurally coherent 3D models. (b) Architectural renovation: identifying and reconstructing modified areas while preserving design semantics.

1.3.2 Challenges in Architectural Design

At the design level, as shown in Figure 1.3, current generative design systems face two main challenges: the generation of new buildings and the renovation of existing ones.

(1) Architectural Generation. The central issue lies in transforming abstract design inputs, such as sketches, into three-dimensional models that conform to structural logic. Sketches reflect the architect's intent and conceptual reasoning, yet their sparse and ambiguous information makes it difficult for models to infer spatial hierarchy and compositional structure. Enabling generative methods to derive depth, structure, and functional organization from sketches remains one of the fundamental challenges of intelligent architectural generation.

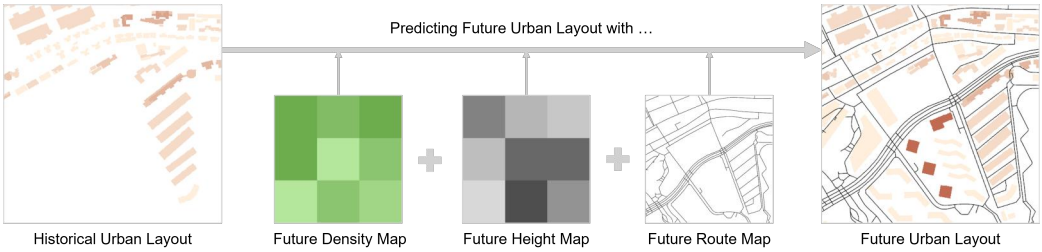


Figure 1.4: Challenges in urban planning and forecasting. Urban layout prediction requires the integrated consideration of multiple factors, such as historical layout information, building height and density, and road network structure.

(2) Architectural Renovation. The difficulty lies in recognizing and reconstructing modified areas while preserving the semantics of the original design. Existing buildings often contain damaged elements, added structures, or multi-phase repairs, requiring models to reason across LoD hierarchies and maintain consistent semantic representation across different levels of precision and stages of design.

To support this direction, this research constructs the LoD Sketch Dataset, encompassing multiple layers of data, including sketches, massing models, and detailed models. This multi-level dataset enables generative methods to learn the hierarchical evolution of architectural forms.

In summary, the challenges in architectural design go beyond geometric generation. They involve semantic understanding and hierarchical reasoning. Generative methods must learn design logic from conceptual expressions and infer spatial intent from geometric structures, thereby establishing a continuous computational framework that bridges creation and renovation.

1.3.3 Challenges in Urban Planning and Forecasting

When generative modeling is extended to the urban level, the challenges become substantially more complex (Figure 1.4). Cities are composed of multiple spatial layers, including buildings, roads, green spaces, and water systems, together with multidimensional attributes such as density, height, and neighborhood relationships.

Unlike static architectural forms, urban spaces exhibit high dynamism and temporal evolution. Most existing urban generation methods [20, 21] focus on static reconstruction, producing a plausible spatial layout under fixed conditions without modeling the temporal dimension. Such approaches fail to capture the continuity of urban development and cannot predict spatial

transformations across different time periods.

In addition, urban-level data are inherently heterogeneous: spatial and temporal variations across regions and years lead to inconsistent data distributions, making it difficult for models to generalize or establish a unified generative logic.

To address these problems, this research proposes a Multi-Modal Diffusion Framework for urban layout forecasting. The framework integrates multiple conditional inputs, including layout, density, height, neighborhood context, and historical time sequences, enabling AI to learn both the spatial organization and dynamic evolution of cities. In this way, the model is capable not only of generating plausible spatial configurations but also of forecasting urban development trends, providing data-driven insights to support forward-looking urban planning and policy decisions.

1.3.4 Summary of Challenges and Research Positioning

Overall, the core challenge of generative modeling in architecture and urbanism lies in achieving unified representation across scales, stages, and time. Existing models are often confined to a single level, such as image-based synthesis or individual building generation, and lack a holistic framework that connects architectural sketches to urban systems.

To overcome these limitations, this study aims to construct a cross-level generative approach that achieves continuity in three dimensions:

1. **From Sketch to Structure:** enabling generative methods to produce 3D architectural forms with structural logic from sketches, thereby bridging conceptual representation and spatial realization.
2. **From Architecture to City:** establishing generative mappings between architectural and urban levels, allowing system to understand spatial organization and systemic structures across different scales.
3. **From Static to Dynamic:** integrating LoD-based architectural evolution with multi-temporal urban data to build a continuous model that unifies spatial generation and temporal prediction.

Together, these three objectives define the overarching goal of this research: to enable generative methods not only to produce architectural forms but also to understand spatial logic and evolutionary patterns, thereby bridging the gap between static representation and temporal forecasting. Through this cross-level and cross-temporal generative framework, the study seeks to lay a theoretical and technical foundation for Intelligent Co-Generative Design, in which architectural and urban systems evolve collaboratively through computational reasoning and synthesis.

1.4 Methodological Framework

To address the challenges outlined above, this research develops a cross-level, multi-modal, and generative AI–driven framework for architectural and urban modeling. The framework integrates Level of Detail (LoD) hierarchies, multi-modal fusion, and generative modeling mechanisms. Its core concept lies in employing representation learning, conditional control, and spatiotemporal modeling to ensure semantic coherence and structural continuity across scales, from conceptual sketches to city-level prediction, thereby establishing an interpretable, verifiable, and extensible generative system.

1.4.1 Research Framework

The overarching objective of this study is to construct a unified generative system capable of bridging different spatial scales and design phases. At the architectural level, the model is designed to comprehend the semantic hierarchy and LoD structure of buildings; at the urban level, it aims to capture large-scale spatial patterns and temporal evolution.

Accordingly, this research proposes a multi-conditional generative framework that uses diffusion models as the core generative engine, ControlNet as the control mechanism, and Large Vision–Language Models (VLMs) as the semantic hub. Together, these components allow the model to generate spatially coherent results guided by multi-modal inputs.

Unlike conventional geometry-driven or parametric modeling approaches, this framework emphasizes semantic consistency and interpretability. Through multimodal learning, the model generates spatial forms while capturing the underlying organizational principles, design semantics, and contextual constraints governing their formation. Structurally, the framework follows a progression from individual form generation to system-level prediction, from local synthesis to contextual continuity, and from spatial geometry to temporal evolution, unifying generative processes and reasoning within spatial modeling.

In essence, the goal is not to pursue visual fidelity alone but to enable AI to participate in design reasoning and spatial expression through structured generative understanding.

1.4.2 Research Tasks and Modules

To systematically validate the framework, five core research tasks are defined, each corresponding to a specific spatial generation level (Figure 1.5).

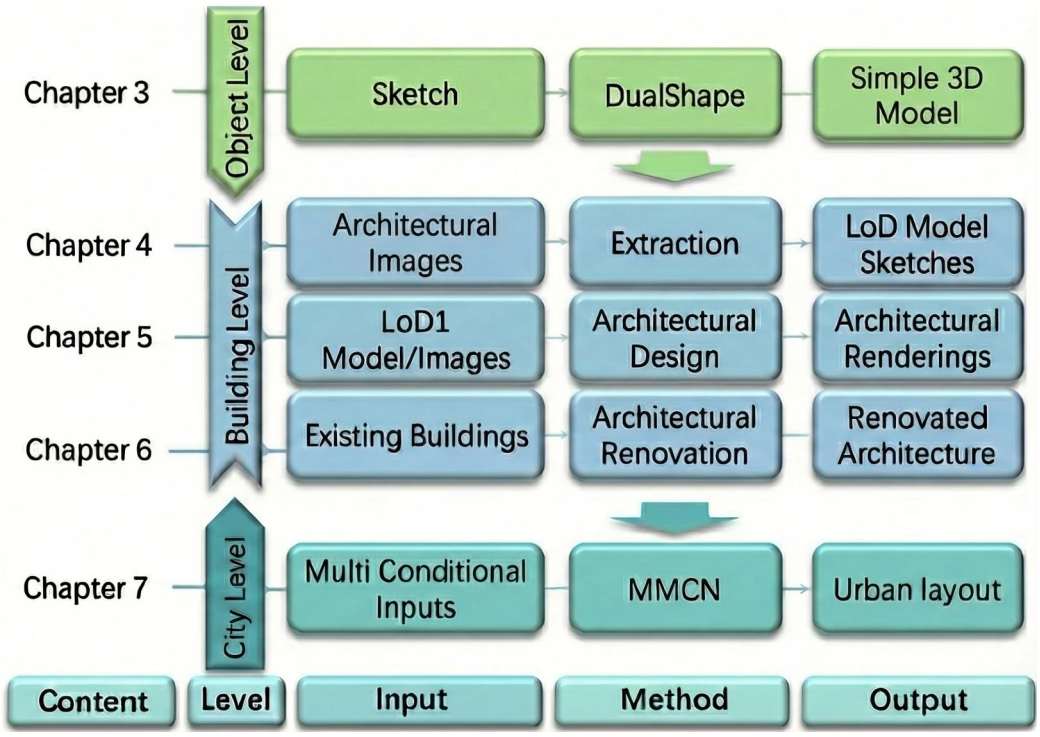


Figure 1.5: Research tasks, modules, and interrelationships.

Together, they form a progressive chain from form to structure to system, representing the logical hierarchy of the study.

1. **DualShape [22]:** This research investigates part-level 3D generation and retrieval from architectural sketches, focusing on the correspondence between geometry and semantics at the component level—forming the foundation of form-level understanding.
2. **LoD Sketch Dataset Construction [23]:** This research builds a cross-LoD multi-modal dataset encompassing sketches, massing models, depth maps, and semantic labels, supporting multi-level training and evaluation of generative models.
3. **Multiview Consistent Architectural Design [24]:** This research addresses the problem of consistency in multi-view architectural generation. By integrating geometric projection constraints and view-alignment mechanisms, the model ensures spatial coherence and structural plausibility across viewpoints.
4. **Architectural Renovation [25]:** This research explores semantic regeneration and automatic updating of existing buildings. By learning mapping relationships between original and updated structures, the

model identifies modifiable regions and achieves semantically consistent renovation generation.

5. **Urban Layout Forecasting [26]:** This research extends the study to the urban level, combining multi-modal inputs and temporal diffusion modeling to learn spatial organization and temporal evolution, enabling multi-period urban layout prediction (e.g., 2005–2024).

These five tasks are hierarchically connected rather than independent (Figure. 1.5). Their interrelations can be described as follows:

- **Object Level:** *DualShape* establish the foundation for AI's understanding of architectural geometry and semantics, providing data and structural bases for higher-level generation.
- **Building Level:** *LoD Sketch Dataset*, *Multiview Generation* and *Architectural Renovation* focus on holistic architectural generation and updating, validating the model's consistency and reasoning ability in complex spatial semantics.
- **City Level:** *Urban Layout Forecasting* extends the framework to urban systems, integrating spatial and temporal dimensions to construct predictive models of urban evolution.

This hierarchical organization embodies a cross-level generative logic that progresses from form to structure and further to system. The model begins with individual form generation, advances to architectural-scale spatial reasoning, and ultimately supports temporal prediction at the urban level. Through this continuous modeling chain, design logic is preserved and propagated across levels, enabling genuinely cross-level generative modeling.

1.4.3 Methodological Significance

The proposed framework carries both technical and academic significance.

First, it establishes a cross-level generative mechanism that conceptualizes architecture and the city as a continuous spatial system. By leveraging shared semantic representation, it enables continuous generation from sketches to urban layouts, providing a novel linkage between Building Information Modeling (BIM) and large-scale urban data.

Second, the framework proposes an interpretable and controllable generative logic, where semantic fusion and spatial memory mechanisms ensure that the generation process adheres to architectural reasoning and spatial coherence, enabling explainable design generation rather than purely data-driven image synthesis.

Finally, the study develops an assisted design validation platform that integrates multi-level tasks and datasets to empirically demonstrate the feasi-

bility of cross-level generative modeling. This platform not only substantiates the theoretical model but also illustrates the potential of generative methods as effective collaborators in architectural design and urban planning.

In summary, the framework extends generative methods from image to spatial production and advances the integration of architectural design and urban planning. It addresses the earlier challenges and provides a solid foundation for research on architectural generation and urban forecasting.

1.5 Structure of the Dissertation

In Chapter 2, the dissertation reviews foundational studies on sketch-based modeling, three-dimensional geometric representations, diffusion-based generative models, and applications of artificial intelligence in architecture and urban design. These discussions clarify the methodological basis of the proposed framework and highlight the limitations of existing approaches that the subsequent chapters address.

In Chapter 3, a sketch-guided and part-aware three-dimensional generation and retrieval system is introduced. This chapter explains how freehand sketches are interpreted as structural cues for part decomposition and shape representation, establishing the object-level foundation of the dissertation's cross-level generative modeling framework.

Chapter 4 extends the research to the building level by presenting an automatic Level of Detail sketch construction framework. This chapter proposes a multi-stage extraction pipeline that produces aligned contour, massing, depth, and semantic representations, forming a coherent LoD hierarchy suitable for data-driven architectural modeling.

Chapter 5 investigates multi-view geometry-consistent architectural design. Drawing on shoebox inputs, depth information, and multi-view alignment mechanisms, this chapter demonstrates how diffusion-based models can generate architectural imagery that maintains stylistic consistency and geometric coherence across multiple viewpoints.

Chapter 6 advances the framework toward architectural intervention through sketch-based and text-driven facade renovation. By integrating VLM, generative inference, and ControlNet refinement, this chapter proposes a renovation pipeline that identifies modifiable regions, synthesizes new facade components, and preserves the structural logic of the existing building.

Chapter 7 extends the generative modeling framework to the urban level through a multi-conditional approach to forecasting urban evolution. By incorporating layout, density, height, road networks, and historical sequences, the chapter presents a diffusion-based model capable of capturing both

spatial organization and temporal development patterns.

Finally, Chapter 8 summarizes the cross-level contributions of the dissertation and discusses remaining challenges and future research directions, including performance-aware generative design, integration with simulation workflows, and the development of autonomous AI-based design agents.

Chapter 2

Related Work

This chapter reviews prior studies that form the conceptual and technical foundations of this dissertation. The discussion is structured along four methodological dimensions: sketch-based representation and modeling, 3D geometric representations, diffusion-based generative mechanisms and control, and AI-driven applications in architecture and urban design. This structure clarifies how seemingly heterogeneous research threads converge toward a unified cross-scale generative framework, and it highlights where the proposed methods in Chapters 3–7 extend or reconfigure existing paradigms.

2.1 Sketch-Based Content Creation

Sketch-based representations constitute an essential modality in visual computing, serving as a concise yet expressive abstraction of geometric intent. In early-stage design, particularly in architecture, product design, and 3D modeling, sketches provide a cognitive bridge between conceptual ideation and computational formalization. Their sparse line-based nature distills complex geometry into semantically meaningful structures, enabling downstream tasks such as image synthesis, part retrieval, and 3D reconstruction.

To situate the contributions of this dissertation, this section reviews three major research directions: (1) sketch-to-image translation, (2) sketch-based retrieval, and (3) sketch-driven 3D reconstruction. For each paradigm, the discussion focuses on methodological evolution, underlying assumptions, and remaining limitations that motivate the multi-scale generative modeling framework proposed in the subsequent chapters.

2.1.1 Sketch-to-Image Translation

Sketch-to-image translation aims to map sparse line drawings to photorealistic images while preserving structural intent. Classical edge-based preprocessing methods, such as Holistically-Nested Edge Detection (HED) [27] and Richer Convolutional Features (RCF) [28], provided the first robust

algorithms for extracting stroke-like boundaries at multiple receptive fields. These methods facilitated sketch extraction from photographs and shaped the representation conventions used in learning-based generative pipelines.

Deep neural models extended these ideas to conditional synthesis. Pix2Pix [29] demonstrated that paired sketch–photo datasets enable translation via adversarial learning, while CycleGAN [30] introduced unpaired translation for settings where sketch–image pairs are infeasible to obtain. However, GAN-based models often struggle with geometric consistency due to the one-to-many mapping inherent in sparse-to-dense translation.

The advent of diffusion models marked a significant shift. Latent Diffusion Models (LDMs) [7] compress images into a latent space where denoising is more efficient while preserving semantic fidelity. ControlNet [8] further formalizes spatially aligned conditioning by injecting encoded sketches, Canny edges, or depth fields directly into a frozen U-Net backbone.

These developments have profound implications for design workflows, in which sketches encode massing, facade layout, and spatial hierarchy. Line-based conditioning thus forms the technical foundation for the multi-Level of Detail (LoD) sketch synthesis and renovation modeling frameworks introduced in later chapters.

2.1.2 Sketch-Based Shape Retrieval

Sketch-based retrieval aims to identify structurally or visually similar shapes from large databases based on freehand sketches. Early approaches relied on handcrafted descriptors and multi-view 2D projections of 3D shapes, using contour similarity to measure correspondence between sketches and rendered silhouettes [31–33]. These systems conceptualized sketch–shape matching as a feature alignment problem: mapping 2D stroke inputs to projected 2D shape cues.

Deep metric learning significantly improved retrieval robustness. Sketch-a-Net [34] and triplet-based embedding networks [35] learned unified latent spaces in which sketches and rendered contours share semantic neighborhoods. Nevertheless, these methods focused primarily on holistic object similarity and lacked mechanisms for interpreting internal structure.

The introduction of hierarchical datasets such as PartNet [36] expanded the retrieval paradigm toward part-aware matching. By modeling part-level relationships, retrieval systems could identify functional subcomponents, such as windows, doors, and roofs, based on incomplete or abstract sketches.

These principles influence this dissertation's modeling philosophy. In particular, Chapter 3 will discuss how sketches serve both as retrieval queries

and generative constraints, enabling hybrid part-based assembly guided by implicit neural representations.

2.1.3 Sketch-Based 3D Reconstruction

Translating sketches into 3D geometry is inherently underconstrained, as line drawings represent a sparse projection of full volumetric form. Early interactive systems such as Teddy [37], SmoothSketch [38], and FiberMesh [39] introduced geometric priors, including inflation, network curves, and elastic surface optimization, to infer plausible 3D surfaces from coarse strokes. These systems highlighted two central principles. First, sketches encode coarse topology but require strong priors to recover detailed geometry. Second, global structural coherence must be maintained to compensate for the sparsity of the input.

Deep learning approaches expanded these capabilities. Volumetric reconstruction methods [40, 41] predict voxel or Truncated Signed Distance Function (TSDF) grids directly from sketch inputs, but resolution limitations and the lack of structural semantics hinder their use in design domains. Multi-view approaches [42] estimate depth or normal maps from sketches across canonical viewpoints, but struggle with topological ambiguities.

Implicit neural representations, such as DeepSDF [43] and Occupancy Networks [44], provide continuous and high-fidelity shape fields. However, inferring continuous signed distance fields (SDFs) from sketches remains challenging due to sparse geometric cues, lack of depth ordering, ambiguous correspondence between strokes and surface patches, and complex hierarchical structures in architectural geometry. Hybrid structured generative approaches (e.g., GRASS [45], StructureNet [46]) combine part hierarchies with generative reasoning, partially addressing these challenges.

These insights motivate the modeling strategies adopted in this dissertation. DualShape [22] (Chapter 3) integrates sketch cues, part retrieval, and implicit field modeling to produce structurally consistent 3D models. Furthermore, the architectural LoD pipeline (Chapter 4) builds upon sketch abstraction principles to generate multi-level sketch representations aligned with 3D structure.

2.2 3D Representation and Modeling

3D representations are the substrate on which generative models operate. Their choice determines how easily shapes can be edited, how well continuity is preserved, and how naturally structural semantics can be encoded. This

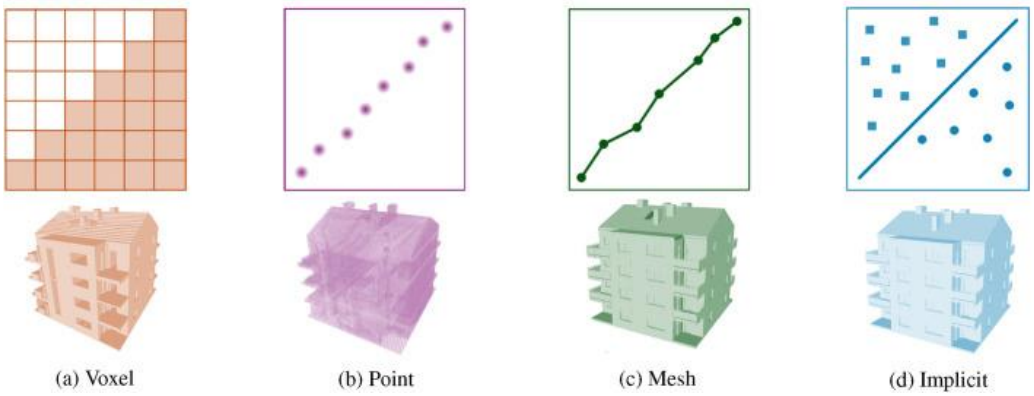


Figure 2.1: Representation examples of 3D shapes [47].

section reviews explicit (mesh, voxel, point cloud) and implicit neural representations, with an emphasis on part-aware modeling and level-of-detail (LoD) abstraction relevant to architecture and urban design. Figure 2.1 illustrates these four representative forms.

2.2.1 Explicit 3D Representations

Meshes, voxels, and point clouds are the most widely adopted explicit representations in graphics and geometric processing. Polygonal meshes, composed of vertices, edges, and faces, are the default representation for rendering, simulation, and CAD modeling. They support efficient visualization, physical simulation, and precise geometric operations. Mesh-based neural architectures such as neural mesh renderers [48] and AtlasNet [49] learn to generate mesh surfaces directly, but they typically require fixed topology or pre-defined parameterization.

Voxel grids discretize space into regular cells and have been used extensively in early 3D deep learning. Volumetric CNNs [40, 50] and occupancy-based networks process voxels with 3D convolutions, enabling straightforward architecture extensions from 2D to 3D. However, voxel resolutions scale cubically with grid size, leading to high memory costs. This is particularly problematic for detailed architectural elements, where window frames, rails, or decorative features require fine resolution.

Point clouds provide an alternative by representing shapes as unstructured sets of 3D points. PointNet and its variants [51, 52] introduced permutation-invariant architectures for classification and segmentation. Generative models such as PSGN [53] and point-flow methods treat point clouds as flexible shape carriers. Nonetheless, the absence of explicit connectivity

complicates surface reconstruction and the enforcement of structural relationships between architectural components.

In architectural practice, explicit representations are tightly integrated into Building Information Modeling (BIM) workflows and 3D city models. Standards such as CityGML [54] and BIM handbooks [13] emphasize the need for multi-LoD models that are both geometrically consistent and semantically annotated. These requirements motivate the hybrid strategies adopted in this dissertation: explicit meshes are retained where high-quality parts exist (e.g., retrieved components), while implicit fields and sketch abstractions handle geometric interpolation and LoD transformation.

2.2.2 Implicit Neural Representations

Implicit neural representations describe shapes as continuous fields, such as signed distance functions (SDFs) or occupancy probabilities, rather than discrete surface elements. DeepSDF [43] showed that a neural network can learn a latent code and SDF mapping that represent entire shape families, enabling smooth interpolation and high-resolution surface reconstruction. Occupancy Networks [44] similarly learn a function that predicts whether a given 3D point lies inside or outside the object.

These methods offer several advantages: resolution-free detail (limited only by evaluation density), natural handling of topology changes, and differentiability for gradient-based optimization and inverse problems. Follow-up work incorporated differentiable rendering [55] and multi-view supervision to improve detail and consistency.

However, implicit fields require explicit surface extraction (e.g., via Marching Cubes [56]), which is not inherently differentiable. MeshSDF [57] addressed this by introducing differentiable mesh extraction from signed distance fields, allowing end-to-end optimization that jointly refines the implicit representation and the resulting mesh. This connection between continuous fields and explicit surfaces is particularly valuable for design workflows that require editable meshes.

In this dissertation, implicit SDFs are used as a flexible backbone for part-level geometry generation. When a relevant component cannot be found in the retrieval database, the system uses sketch-conditioned SDF modeling to synthesize compatible geometry. This approach combines the geometric expressiveness of implicit fields with the structural reliability of part-aware retrieval.

2.2.3 Part-Aware Modeling and Semantic Decomposition

Many man-made objects and buildings exhibit hierarchical structure: facades are composed of floors, bays, and openings; furniture includes legs, seats, and backrests; vehicles have bodies, wheels, and accessories. Part-aware modeling exploits this structure by decomposing shapes into semantic components and modeling their relationships explicitly.

The PartNet dataset [36] provides fine-grained, hierarchical segmentations for thousands of 3D models across multiple categories, enabling data-driven research on part decomposition, relational modeling, and assembly. StructureNet [46] and related graph-based generative models learn to encode shape structure as graphs of parts and connections, supporting both unconditional shape generation and structure-aware editing. Im2Struct [58] explores similar ideas in the context of RGB-D indoor scenes, recovering object hierarchies from images.

Assembly-based modeling frameworks go one step further by retrieving and composing parts from existing databases. Li et al. [45] and other systems demonstrate that reusing high-quality components can yield more realistic geometry than purely generative decoders, especially for detailed man-made objects. Such methods often rely on learned compatibility metrics and relational constraints to ensure coherent assemblies.

The part-aware perspective directly underpins the DualShape framework in Chapter 3. Rather than treating sketch-based modeling as a monolithic mapping from strokes to shape, the proposed method decomposes objects into instance parts, leverages sketch-based retrieval for components with strong priors, and uses SDF-based generation for novel or underrepresented parts. This combination reflects a broader trend in 3D modeling: from holistic shape decoding toward structure-aware, modular, and reconfigurable generative systems.

2.2.4 Shape Abstraction and Level of Detail

Level of Detail (LoD) modeling is crucial for both graphics and urban information systems. In 3D city models, LoD levels determine which geometric and semantic details are present at each scale, from simple building footprints to full facades with openings and roof structures. CityGML formalizes several LoD levels (e.g., LoD0–LoD3) and their intended use cases [54]. Biljecki et al. [15,59] systematically analyzed LoD definitions, their interpretation across software environments, and their impact on analysis accuracy.

Traditional LoD simplification techniques focus on geometric reduc-

tion, such as mesh decimation, planar segmentation, and feature-preserving remeshing. While effective for visualization, these methods often lack semantic awareness: the relationship between, for example, windows in a LoD3 model and corresponding wall surfaces in LoD2 is not explicitly maintained. This makes it difficult to reason about consistency across levels or to train learning-based models that operate on multiple LoDs.

More recent studies have started to address semantic LoD alignment, but they are typically constrained to small datasets or specific building types. A significant bottleneck remains the scarcity of large-scale, geometrically aligned multi-LoD datasets with explicit inter-level correspondences between features such as walls, roofs, and openings.

Chapter 4 contributes to this gap by introducing an Automatic LoD Sketch Extraction pipeline and associated dataset. By starting from high-detail 3D models and systematically generating aligned LoD1–LoD3 sketches across multiple views, the dataset provides a standardized foundation for LoD-aware generative modeling. This resource enables diffusion models to learn how structural details emerge or disappear across levels and provides a reproducible benchmark for evaluating LoD transitions in architectural generation.

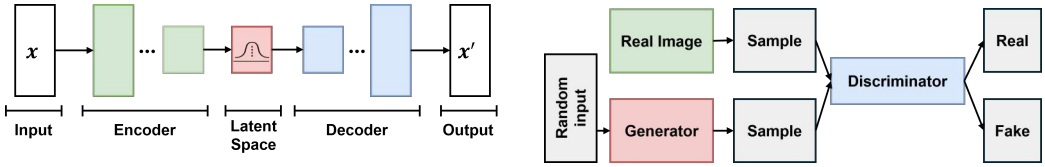
2.3 Generative Models

The third pillar of this dissertation is the family of generative models that translate multi-modal inputs into images, sketches, or layouts. While early work focused on VAEs and GANs, diffusion models have recently become the dominant paradigm for high-fidelity image synthesis. This section reviews key generative frameworks and the evolution of conditioning mechanisms that enable fine-grained control.

2.3.1 Generative Model Paradigms

Variational Autoencoders (VAEs) [12] introduced a probabilistic encoder–decoder framework that learns a latent distribution over data. VAEs optimize a variational lower bound on the data likelihood and allow efficient sampling from a continuous latent space. However, the reconstructions produced by VAEs are often blurry, which limits their usability for detailed architectural or urban imagery.

Generative Adversarial Networks (GANs) [5] formulate generation as a minimax game between a generator and a discriminator. GANs have produced impressive results in image synthesis, super-resolution, and style trans-



(a) VAE (b) GAN

Figure 2.2: Overall pipeline of two generative models: (a) VAE and (b) GAN.

fer, including applications to floorplan [60, 61] and facade generation [62]. Nevertheless, GAN training is notoriously unstable and prone to mode collapse, especially when modeling diverse distributions such as urban forms. As shown in Figure 2.2, VAEs and GANs represent two foundational generative paradigms with distinct training dynamics and structural constraints.

Normalizing flows [63] take a different approach by learning invertible transformations between simple base distributions and complex data distributions, allowing exact likelihood computation. While theoretically elegant, their architectural constraints and computational demands have limited their adoption in large-scale image synthesis.

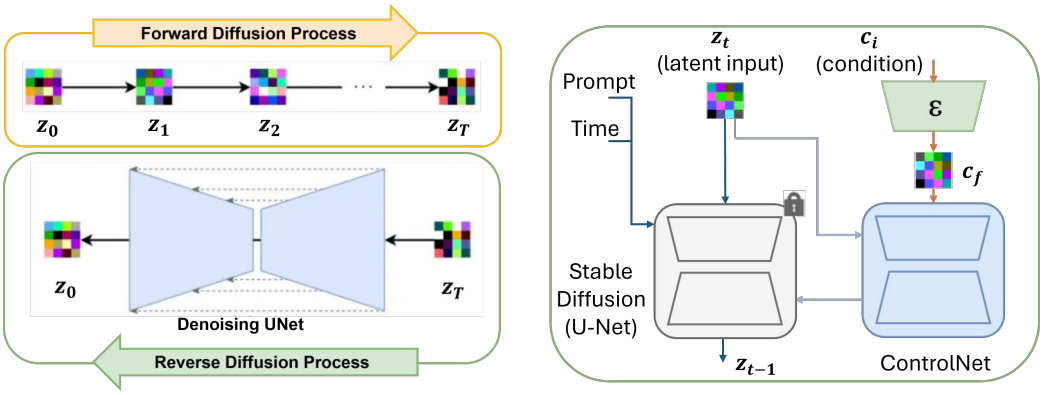
Diffusion models, exemplified by Denoising Diffusion Probabilistic Models (DDPMs) [6] and subsequent score-based formulations [64], simulate a Markovian diffusion process that gradually converts noise into data. They have shown superior diversity and fidelity compared to GANs across many benchmarks. Latent Diffusion Models (LDMs) [7] further improved efficiency by performing diffusion in a compressed latent space, enabling high-resolution generation with manageable computational cost.

Chapters 4–7 build on LDM-based architectures as the main generative backbone. Their stability, compatibility with rich conditioning, and strong reconstruction quality make them well-suited for multi-LoD sketch generation, facade renovation, and multi-conditional urban evolution forecasting.

2.3.2 Text-to-Image Generation

Text-to-image generation aligns natural language prompts with images. CLIP [9] first demonstrated that large-scale contrastive pre-training on image–text pairs can learn a joint embedding space that supports zero-shot classification and retrieval. Building on such embeddings, several text-to-image diffusion models have been proposed.

GLIDE [65] used classifier-free guidance and a cascaded architecture to generate images from text descriptions, while DALLE-2 [66] combined CLIP-based priors with diffusion to produce high-resolution, semantically aligned



(a) Latent Diffusion. (b) Stable Diffusion with ControlNet

Figure 2.3: (a) Latent Diffusion: the model learns to denoise latent variables from $\mathbf{z}_T$ to $\mathbf{z}_0$ through a U-Net. (b) Stable Diffusion with ControlNet: encodes condition input $\mathbf{c}_i$ into guidance features to modulate the denoising of latent input $\mathbf{z}_t$.

images. Imagen [67] emphasized the importance of powerful text encoders and scaling, achieving state-of-the-art quality in text-to-image benchmarks. Stable Diffusion [7] popularized a practical LDM-based framework that is open and extensible, making it a de facto platform for many downstream applications.

For architectural and urban design, text-to-image diffusion offers an intuitive means of specifying style, usage, and environmental conditions (e.g., "brick facade with large industrial windows at dusk"). However, architectural geometry is tightly constrained by structural logic and regulatory requirements, which are difficult to guarantee through text alone. This motivates the integration of textual prompts with sketches, depth maps, semantic segmentations, and historical layouts as additional, more precise controls, an approach adopted in this dissertation.

2.3.3 Control in Diffusion Models

As shown in Figure 2.3, latent diffusion enables efficient denoising in the compressed latent space, while ControlNet augments the diffusion process with structural conditioning through an additional trainable branch.

Controllability is central to deploying diffusion models in design workflows. Classifier guidance and classifier-free guidance are two key techniques for conditional diffusion. Dhariwal and Nichol [68] showed that adding classifier gradients during sampling can steer diffusion toward specific classes,

while classifier-free guidance [69] simplifies this by jointly training conditional and unconditional models and combining their outputs at sampling time.

Beyond label conditioning, structural controls such as edges, poses, and depth maps have become important. ControlNet [8] introduced a practical architecture in which a frozen text-to-image backbone is augmented with a trainable control branch that processes additional input maps. By preserving the parameters of the original model, ControlNet maintains pre-trained semantics while enabling precise geometric alignment with control signals. T2I-Adapter [70] further explored lightweight plug-in adapters that inject control features into a frozen diffusion backbone without duplicating the entire network. Composer [71] demonstrated compositional conditioning, allowing multiple independent conditions (e.g., layout, style, and color) to be combined in a single generation process.

These mechanisms provide general templates that the proposed LoD sketch generation and MMCN frameworks build upon. In Chapter 4, sketch-, depth-, and mask-like controls are used to enforce geometric consistency across LoD levels and views. In Chapter 7, multiple urban factors (density, height, road networks, and historical layouts) are encoded as distinct conditional branches and fused via multi-prompt and memory-based mechanisms, extending existing control paradigms toward spatio-temporal urban modeling.

2.4 Architecture and Urban Design

The final dimension of related work concerns applications of generative AI to architecture and urban planning. Compared to generic image domains, built environment applications impose stricter constraints on structure, continuity, and interpretability. This section reviews representative methods for architectural form generation, facade and scene modeling, and urban layout synthesis, and clarifies how this dissertation extends these lines toward multi-LoD and temporally coherent generative frameworks.

2.4.1 Architectural Form and Layout Generation

Early AI-based form generation in architecture often relied on grammars, shape rules, and procedural modeling. While flexible, these systems required manual rule design and were difficult to calibrate using real-world data. The advent of deep generative models brought data-driven approaches to room and floorplan layout. For instance, House-GAN [72] used relational GANs to

model room arrangements and connectivity, while other works applied CNNs and GANs to floorplan vectorization and synthesis.

At larger scales, BlockPlanner [73] used VAEs to generate urban block layouts that encode both building footprints and land-use information, and Johanes and Huang [74] proposed a Generative Isovist Transformer that generates spatial sequences optimized for visibility and accessibility. These studies demonstrate that generative models can capture meaningful spatial patterns, but they often focus on a single scale (e.g., building or block) and a single snapshot in time.

Diffusion-based models have also been explored for layout generation. Recent work on diffusion for floorplan synthesis and room topology [75] shows that diffusion can capture complex structural relationships and improve diversity over GAN baselines. However, most of these models operate on schematic layouts or raster representations rather than directly on multi-LoD architectural sketches or city-scale evolution sequences.

The methods in Chapters 4, 5, and 7 build on these insights by explicitly modeling cross-scale and temporal dimensions. Chapter 4 constructs an automatic multi-Level of Detail sketch representation that supports LoD-aware architectural modeling. Chapter 5 develops a diffusion-based framework for multi-view geometry-consistent architectural design from simplified volumetric inputs. Chapter 7 employs a multi-conditional diffusion architecture to forecast urban evolution under multiple interacting factors.

2.4.2 Generative Facade Renovation

At the facade and streetscape levels, generative methods have been used to synthesize textures, procedural structures, and stylistic variations. Early work combined procedural grammars with example-based texture synthesis to create plausible facades, but required hand-authored rules and lacked flexibility across typologies. More recent approaches leverage image-to-image translation and diffusion models to propose facade variants directly from sketches, labels, or photographs.

Pix2Pix-style methods have been applied to map semantic labels or edges to facade textures, enabling controllable editing and rapid style variations. Diffusion-based models conditioned on edge maps or depth can similarly generate realistic facades while preserving building outlines [7, 8]. IP-Adapter-style techniques [76] introduce reference images as additional conditioning, enabling the transfer of stylistic attributes from exemplar facades to target structures.

Despite these advances, most facade-related generative models treat existing buildings as static canvases, focusing on aesthetic variation rather

than structural renovation. They often ignore constraints such as load-bearing elements, existing openings, or industrial reuse scenarios. Moreover, they rarely operate within a systematic LoD framework: sketches, detailed renderings, and semantic annotations are treated as separate artifacts rather than as different representations of the same underlying building.

Chapter 6 addresses these gaps by proposing a three-stage facade renovation framework that integrates a vision–language model for structural interpretation, a diffusion model for component generation, and ControlNet-like conditioning for photorealistic rendering. Rough structural sketches and textual descriptions are translated into consistent renovation proposals that preserve key geometric constraints while exploring diverse design alternatives. The method can be seen as a domain-specific adaptation of sketch- and subject-guided diffusion models, tailored to the needs of adaptive reuse and renovation workflows.

2.4.3 Urban Layout Generation

Generative modeling at the urban scale poses additional challenges: spatial patterns extend across large areas, different factors (density, height, transportation networks, zoning) interact in complex ways, and urban evolution unfolds over long time horizons. Several lines of work address parts of this problem.

GAN-based frameworks such as Urban-GAN [21] use adversarial learning to synthesize urban layouts or street patterns from remote-sensing data, sometimes incorporating participatory design elements. BlockPlanner [73] and related models generate static urban blocks that are functionally consistent and visually plausible. More recently, CityDreamer [77] and CityGen [78] employ compositional diffusion strategies to generate unbounded, semantically coherent city-scale layouts by combining background environments with building instances.

Parallel to generative approaches, traditional urban evolution modeling uses machine learning and statistical methods such as Random Forests, Support Vector Machines (SVMs), and Artificial Neural Networks (ANNs) to predict land-use change or urban expansion from temporal remote-sensing series and socio-economic indicators. These models are effective for forecasting scalar indicators or classification maps but do not generate detailed spatial layouts.

Existing diffusion-based urban generators focus predominantly on static snapshots and often operate on isolated tiles or patches, neglecting both boundary continuity and long-range temporal dependencies. This leads to spatial discontinuities (e.g., broken roads across patch boundaries) and

temporally inconsistent predictions that ignore historical context.

Chapter 7 proposes MMCN (Memory-aware Multi-Conditional generation Network) to address these limitations. MMCN integrates multiple urban factors via multi-conditional control (e.g., density, height, road networks, and historical layouts), incorporates neighbor-aware memory modules to enforce cross-patch continuity, and models temporal evolution by conditioning on historical configurations. In doing so, it extends existing multi-conditional diffusion and urban generation frameworks from static synthesis toward dynamic, spatio-temporally coherent forecasting that is more aligned with sustainable urban planning needs.

Summary

In summary, prior work on sketch-based modeling, 3D representation, diffusion-based generation, and AI-assisted architectural and urban design provides rich foundations but also exhibits several common limitations: limited support for part-aware sketch-controlled 3D modeling, lack of large-scale multi-LoD architectural datasets with consistent semantics, insufficient integration of structural and semantic constraints in facade renovation, and a focus on static snapshots rather than temporally coherent urban evolution.

The subsequent chapters of this dissertation build on and connect these domains. Chapter 3 introduces a part-aware sketch-guided generation and retrieval framework for three-dimensional objects. Chapter 4 constructs an automatic multi-Level of Detail sketch dataset and methods for LoD-consistent architectural modeling. Chapter 5 develops a diffusion-based framework for multi-view geometry-consistent architectural design. Chapter 6 presents a vision–language model-guided, sketch-based and text-conditioned facade renovation approach. Finally, Chapter 7 proposes a memory-aware multi-conditional diffusion framework for urban evolution forecasting. Together, these contributions form a cross-scale generative modeling framework that responds to the gaps identified in this chapter.

Chapter 3

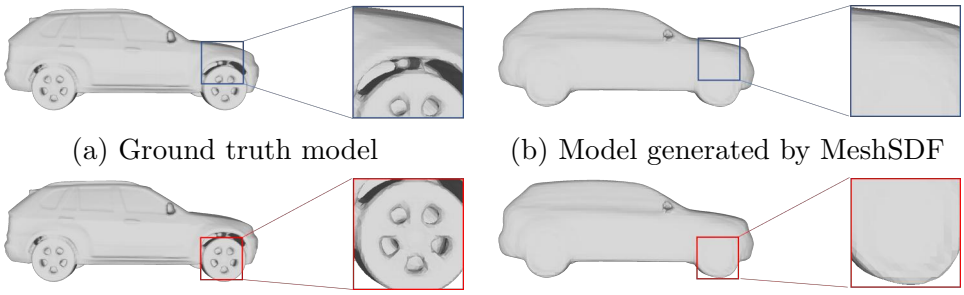
Sketch-Guided 3D Part-Aware Generation and Retrieval

This chapter positions sketch-guided, part-aware 3D shape generation as the object-level component of the dissertation's cross-level generative modeling approach [22]. Focusing on free-hand sketch interpretation, part decomposition, and controllable assembly, it establishes core mechanisms for representing and manipulating 3D geometry in a structurally consistent manner. These mechanisms form the technical and conceptual basis on which the subsequent chapters extend generative modeling from individual objects to building-level multi-LoD representations and, ultimately, to city-level multimodal urban evolution.

3.1 Background

3D shape reconstruction has long been recognized as a core problem in computer graphics. Applications include digital entertainment and architectural modeling. Creating 3D models through conventional commercial software such as Autodesk, AutoCAD and 3ds Max requires substantial manual effort and extensive expertise. Although object scanning provides an alternative means of acquiring geometric models, dedicated equipment is often costly and still faces difficulty in capturing intricate structures. Recent learning based techniques have drawn attention as a promising direction for 3D model generation [79], but large scale training data are often necessary and generated outputs remain constrained by data scarcity and architectural limitations. To address these issues, this study proposes a hybrid modeling strategy that integrates retrieval and generative synthesis in order to support more practical modeling workflows.

The primary motivation is to enable non expert users to construct 3D models from free hand sketches without relying on advanced modeling knowledge. Sketch based reconstruction provides an intuitive interaction modality [41], yet sketches are often sparse, imprecise or ambiguous. These



(a) Ground truth model (b) Model generated by MeshSDF

(c) Ground truth model (d) Model generated by MeshSDF

Figure 3.1: Comparison between the ground truth model and the result produced by MeshSDF [57]. (a) The ground truth exhibits well-defined structural connections between individual components. (b) In the MeshSDF-generated model, these inter-component boundaries appear indistinct. (c) The ground truth tire components display clear geometry and distinctive fine-scale features. (d) The tire details in the MeshSDF output are noticeably smoothed and lack structural clarity.

characteristics introduce difficulties in both interpretation and reconstruction. Template based or component based tools can reduce ambiguity but increase user burden by requiring additional adjustment. Tutorial based systems separate learning from drawing, which prevents correction during sketching. To overcome such limitations, this study introduces a shadow guidance strategy that delivers real time visual cues during sketch creation and supports more structured drawings.

Designing or generating 3D models with fine grained details remains a challenging task. Existing generative approaches such as MeshSDF [57] face difficulties in preserving structural boundaries and detailed geometry as illustrated in Figure 3.1. Using a vehicle model as an example, the outer shell contains many components, including the hood, roof, rear section, doors and structural frame. Creating each element manually is time consuming for non expert users. The large structural variation in car shells makes generative strategies more suitable for these components. In contrast, the tires exhibit high regularity, limited variation and strong symmetry. Drawing tires individually increases user effort since the geometry is smaller and more detailed. Treating the tires as a single component reduces sketching difficulty and retrieval based methods can obtain detailed tire models even from coarse input. Boundaries between major components, such as between the shell and the wheels, are usually clear as shown in Figure 3.1. An assembly based strategy therefore supports the construction of coherent structures with consistent inter part relationships.

Implicit representation research provides important context for these challenges. Implicit 3D shape representations define surfaces as zero level sets of volumetric functions [80]. Neural networks can map coordinates to signed distance values [43] or occupancy probabilities [44], creating continuous and resolution independent surface representations. However, explicit surfaces require repeated sampling within the field which increases computational cost. Many applications require mesh parameterization but extraction procedures such as Marching Cubes [56] lack differentiability. MeshSDF [57] addresses this by introducing a differentiable conversion from distance fields to explicit meshes. DualShape adopts a signed distance based formulation to support generative modeling within the framework.

Another relevant direction is sketch based retrieval and modeling [81]. Early research such as Loffler et al. [82] introduced systems for refining keyword based search results with sketch queries that express intended viewpoints. Funkhouser et al. [31] developed an image driven retrieval system that accepts sketched projections as input. Later research explored more robust sketch descriptors. Ma et al. [83] extracted stroke features from densified sampling along curves and Eitz et al. [33] proposed a Bag of Features representation for sketch based retrieval. Sketches are sparse compared with natural images and carry ambiguities. Systems such as Igarashi et al. [37] convert contour drawings to 3D geometry in real time. Lun et al. [42] and Li et al. [84] infer depth or normal maps before producing 3D geometry, yet fine scale details are often lost because these methods rely on holistic sketches. Other research targets specific domains. Han et al. [85] reconstructed facial geometry from sketches. Nishida et al. [86] used programmatic generation based on inferred parameters. Delanoy et al. [41] constructed volumetric 3D shapes from sketches but voxel resolutions limit detail fidelity. DualShape combines generation and retrieval to mitigate such limitations. High detail components are obtained through retrieval while diverse or structurally variable parts are generated. This balance improves detail richness and modeling flexibility.

Part based modeling and assembly also contribute important insights. Commercial modeling software such as Maya and 3ds Max support part based design, which is difficult for non expert users. Li et al. [87] learned part generation and assembly for structural shape synthesis using volumetric representations, although semantic dependence limits applicability to complex structures. Du et al. [88] decomposed modeling into part generation and assembly. These approaches build upon structural shape learning [46,58] and datasets with dense part partitions [36]. DualShape divides artificial objects into instance parts and performs sketch based retrieval and generation to improve modeling effectiveness.

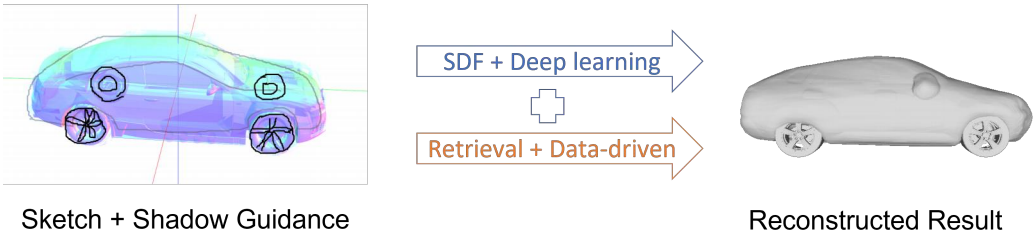


Figure 3.2: Overview of the DualShape framework, which integrates part retrieval and generative modeling to produce hybrid 3D shapes.

Sketch guidance techniques support structured sketch creation. Early systems such as Teddy [37] and the interface in [89] assisted users by correcting low level stroke properties. More advanced strategies retrieve related images and blend them beneath the drawing canvas to provide shadow based guidance [90]. These approaches have proven effective in portrait sketching [91], anime character design [92] and motion retrieval [93]. Limpaecher et al. [94] introduced adaptive correction based on learned stroke variations and Iarussi et al. [95] developed construction line guides for improved accuracy. The present work employs blended three dimensional models to provide shadow cues that assist perception of spatial and structural relationships during sketching.

This study introduces DualShape, a 3D shape design framework that integrates implicit and explicit modeling representations as illustrated in Figure 3.2. The system takes hand drawn sketches as input and leverages signed distance functions for generative synthesis [43] together with retrieval of detailed components. The interface provides real time shadow guidance based on multiple candidate models that align with the drawn strokes. Retrieved and generated components are assembled into complete shapes. Comparative experiments and a user study indicate that DualShape produces geometric detail richer than baseline methods and users consider the system intuitive and easy to operate.

The main contributions of this work are listed as follows:

- A new 3D modeling framework that integrates implicit generative shape representations with shape-based retrieval, forming a hybrid pipeline for model creation.
- A data-driven strategy that performs geometric decomposition and subsequent assembly, enabling the construction of individual model

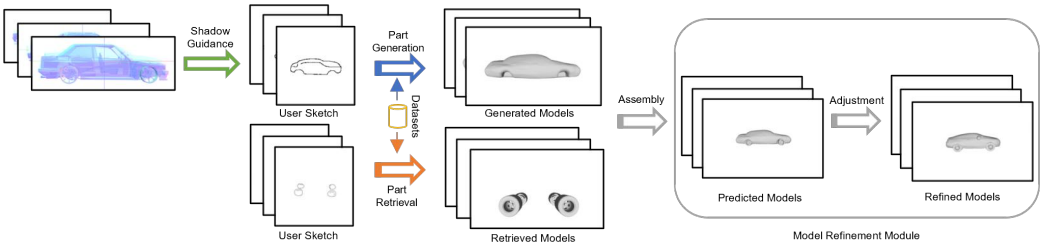


Figure 3.3: Pipeline of DualShape. Hand-drawn sketches serve as input, from which the system retrieves or generates the relevant components. The retrieved and synthesized parts are subsequently assembled into a full 3D model, followed by optional user-driven refinements to obtain the finalized output.

components according to their structural characteristics and design requirements.

- An interactive interface built upon the proposed framework, equipped with real-time design guidance, which allows users without prior modeling experience to efficiently produce 3D shapes.

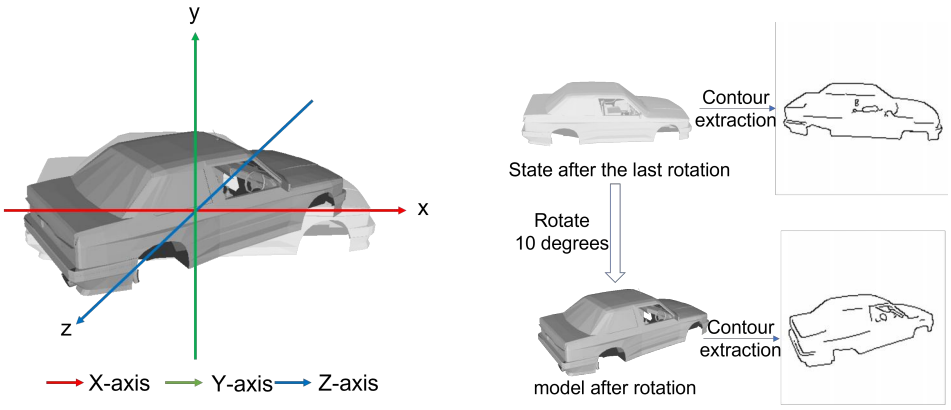
3.2 Methodology

This study presents DualShape, a sketch-driven hybrid framework for 3D shape design. As illustrated in Figure 3.3, the system consists of four primary components: a part retrieval module, a part generation module, a part assembly module, and a model refinement module.

3.2.1 Datasets

In this work, an assembly-based strategy is employed in which complete 3D models are constructed from individual instance-level parts. Since no existing dataset directly satisfies the requirements of this research, both the part dataset and the corresponding contour dataset were created manually by segmenting models according to part annotations and extracting contour images for each resulting instance.

Observations made during dataset construction indicate that dividing objects into meaningful structural components, rather than fragmenting them into a large number of minor sub-parts, leads to more effective modeling. Taking the car category as an example, a typical vehicle model can be decomposed into two main instance-level components: the car shell and



(a) Model state in coordinate system (b) Contours extraction

Figure 3.4: Example of rotating the model to extract contours.

the tires. This hierarchical representation simplifies the generation process by eliminating unnecessary geometric complexity. In addition, separating the model into these two primary components increases the adaptability of each part and facilitates their reuse across different tasks and applications. Overall, modeling a car through its two principal components, the shell and the tires, enhances the clarity of the representation and improves maintainability and reusability.

The part dataset used in this study is derived from the ShapePFCN dataset [96], which contains 3D models collected from diverse online sources. In the original dataset, car models are annotated with four semantic labels: roof, hood, frame, and wheels. For the purposes of this research, these models were reorganized into two higher-level instance components in order to simplify the sketching workflow. The car shell, which includes the roof, the front cover, the rear cover, and the frame, was treated as one component, and the tires were treated as a second component. These were extracted and stored as independent model files. Following this restructuring, the resulting part dataset contains 1,000 instances in total, consisting of 500 car shells and 500 tire models.

To support sketch-based input in the user interface, contour images must be generated from the extracted instance parts. For the part-generation module, and given that user-drawn strokes are typically coarse and simplified, the contours of the car shell were obtained using Canny edge detection. This produces contour representations that are close in style to hand-drawn sketches. As illustrated in Figure 3.4, the car shell contours were generated by placing the model in a canonical orientation where the Y-axis of the 3D

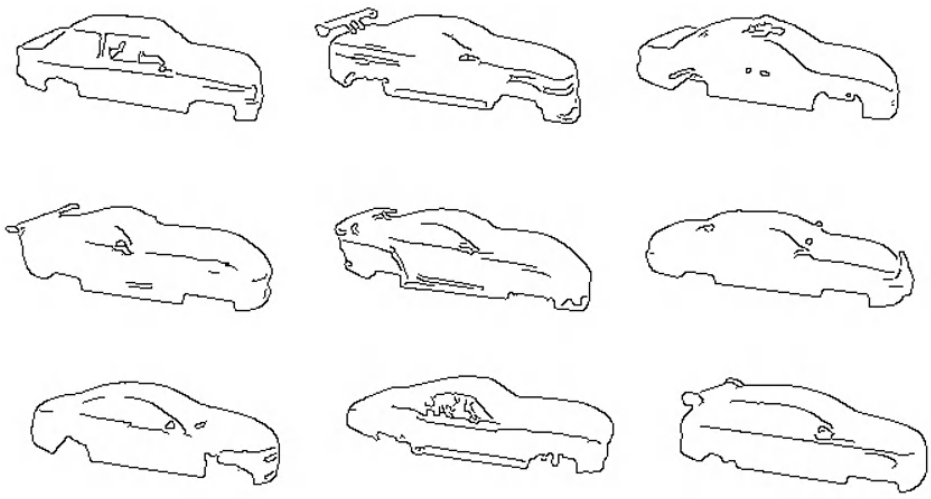


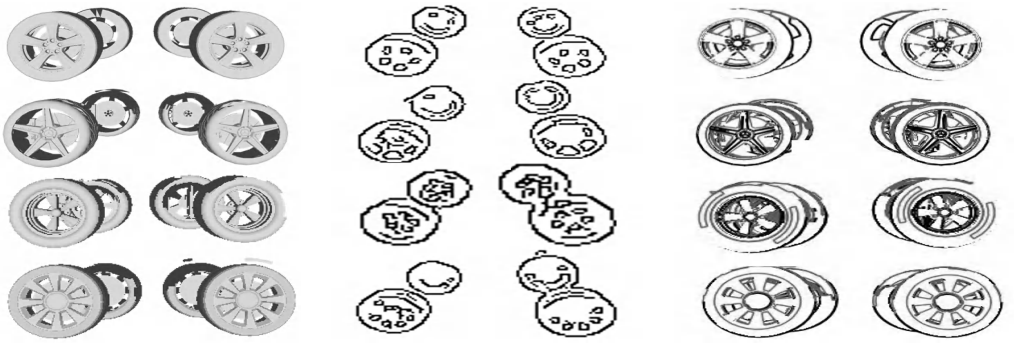
Figure 3.5: Examples from the car shell sketch dataset.

coordinate system (x, y, z) serves as the rotation axis. The model was rotated in increments of 10 degrees, resulting in 36 viewpoints per instance. Through this procedure, a total of 18,000 car-shell contour images were collected for the dataset. Representative examples of the extracted contour data are shown in Figure 3.5.

For the part retrieval module, the contours extracted from the separated wheel models posed a challenge. Because of the limited geometric complexity and the relatively small visual differences among tire contours, Canny edge detection fails to capture distinctive features, making it difficult to retrieve the correct tire model. As illustrated in Figure 3.6b, the contour generated by the Canny operator provides an insufficient representation of the tire's structural characteristics. To overcome this issue, the open-source sketch search engine for 3D object retrieval (OpenSSE) [97, 98] is adopted, as shown in Figure 3.6c. OpenSSE renders models from 102 uniformly sampled viewpoint directions, producing a corresponding set of 102 contour images for each tire. Using this method, a total of 51,000 tire contour images were collected for the dataset. Compared with Canny-based extraction, the contour images generated by OpenSSE preserve the distinctive geometric cues of the tires more clearly, thereby improving the reliability of the retrieval process.

3.2.2 Part Retrieval

To streamline the generation of complete 3D car models, a data-driven retrieval strategy is adopted for the tire components, leveraging their bilateral



(a) car tire (b) tire contour(canny) (c) tire contour(openSSE)

Figure 3.6: Comparison of tire contour extraction methods. (a) The original tire model; (b) the contour obtained using OpenCV's Canny edge detector; (c) the contour generated using the OpenSSE-based extraction approach.

symmetry and the repetitive patterns typically found on tire surfaces. Users only need to sketch the tire tread pattern, from which the system retrieves the most similar tire model and integrates it into the final assembly. The basic functionality of the part retrieval module is illustrated in Figure 3.7. The retrieval process consists of two major stages: (1) encoding contour images in the dataset using a bag-of-features (BoF) representation and constructing the corresponding visual vocabulary, and (2) extracting features from the user's sketch and performing similarity-based search against the vocabulary. Specifically, the BoF framework is applied to the 51,000 contour images in the tire dataset. Interest points are randomly sampled from each image, and a histogram of gradient orientations is computed within the local neighborhood of each selected point. The dominant gradient direction is used to characterize the edge structure around the interest point, and the collection of these descriptors forms the feature bag. In addition, a Gabor local line-based feature (GALIF) filter is employed to extract localized features, enhancing the system's ability to discriminate between subtle variations in tire contours.

The user-provided sketch is first processed to extract its local descriptors, which are then encoded into a compact feature vector. During retrieval, this vector is compared against entries in the preconstructed visual vocabulary to identify the nearest matches. For each sketch, multiple candidate images with varying similarity levels can be obtained by accumulating the matching counts among feature clusters. The tire model corresponding to the most similar retrieved image is subsequently loaded into the user interface. If the retrieved model does not satisfy the user's expectation, the sketch can be modified and the retrieval procedure repeated until a suitable tire model is

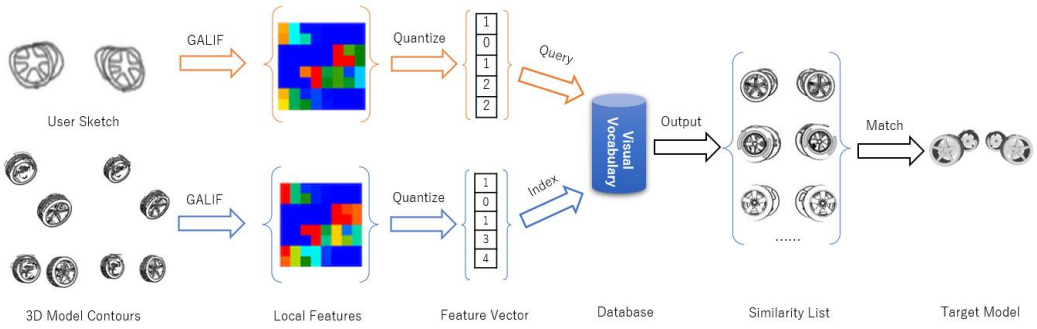


Figure 3.7: Overview of the part retrieval module. The user's sketch is first encoded using GALIF features, which are then matched against the constructed visual vocabulary. The model instance with the highest similarity score is selected as the retrieved part.

obtained.

3.2.3 Part Generation

Generalizing the geometric characteristics of car shells is challenging due to their composite and structurally diverse nature. By leveraging implicit representations such as signed distance functions (SDFs), geometric deep learning techniques enable high-fidelity surface modeling for shapes of arbitrary topology without relying on discretized Euclidean grids, thereby providing a learnable and resolution-independent parameterization. In this work, a differentiable approach is employed for converting SDFs into explicit mesh surfaces, following the MeshSDF framework [57]. The network architecture used for generating component geometries is illustrated in Figure 3.8. Within this formulation, 3D shapes are modeled as the zero-level iso-surfaces encoded by a neural network trained to approximate the SDF. For any spatial point x, the signed distance function, defined as the distance to the nearest surface, is expressed as:

$$SDF(x) = s : x \in \mathbb{R}^3, s \in \mathbb{R} \tag{3.1}$$

The target surface is implicitly encoded as the zero level set of the function satisfying $SDF(\cdot) = 0$. With a sketch provided as input, the goal is for the network to infer an appropriate signed distance field. Accordingly, the optimization objective can be formulated as shown in Equation 3.2.

$$SDF = D(E(S), G) \tag{3.2}$$

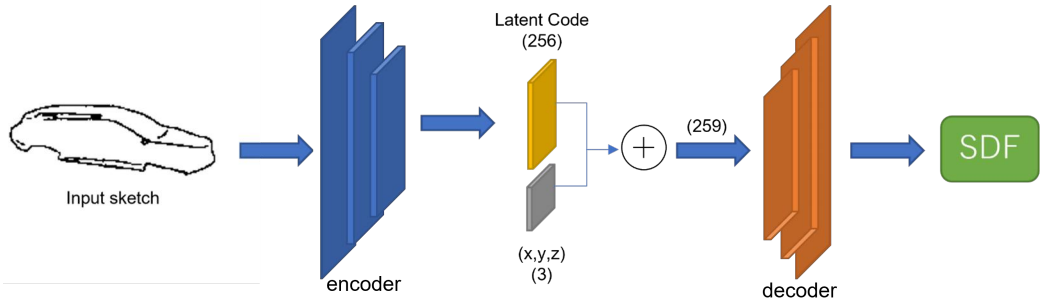


Figure 3.8: Network structure.

Here, S represents the input sketch, and E denotes the encoder that maps the sketch to a latent code z_s. Let G be the set of features associated with the sampled 3D coordinates. The latent code z_s is concatenated with the feature set G to form the latent vector z, which is then fed into the decoder D to generate the corresponding signed distance field. For each sampled coordinate $p_i \in G$, the decoder D outputs a signed distance prediction d_i.

In this implementation, the encoder design follows a structure similar to MeshSDF [57]. A ResNet18 backbone is employed, where the input sketch is combined with a depth-implicit field through a residual image encoder that maps the sketch into a latent code vector. Furthermore, the auto-encoder formulation introduced in DeepSDF [43] is incorporated into the framework. The latent code vectors produced by the encoder are then used to condition the multilayer perceptron (MLP) responsible for predicting the signed distance function.

3.2.4 Part Assembly

The retrieval and generation modules operate solely at the part level, producing suitable 3D shapes without incorporating global structural information. Consequently, the generated components are positioned uniformly at the system's origin (Figure 3.9a), rather than reflecting their actual spatial arrangement in a full vehicle model. To address this issue, the spatial distances and positional correlations between complete car models and their corresponding parts within the dataset were estimated. Based on the geometric centers of the components, the relative positions and scale ratios between the car shells and the tires were computed to restore their correct spatial configuration.

In the initial stage of the assembly module, appropriate placement and alignment of the car shell and tires must be ensured. To accomplish this, a

set of heuristic rules was defined based on prior knowledge extracted from the dataset, enabling the estimation of relative positions and scale relationships between components. The rules were established as follows:

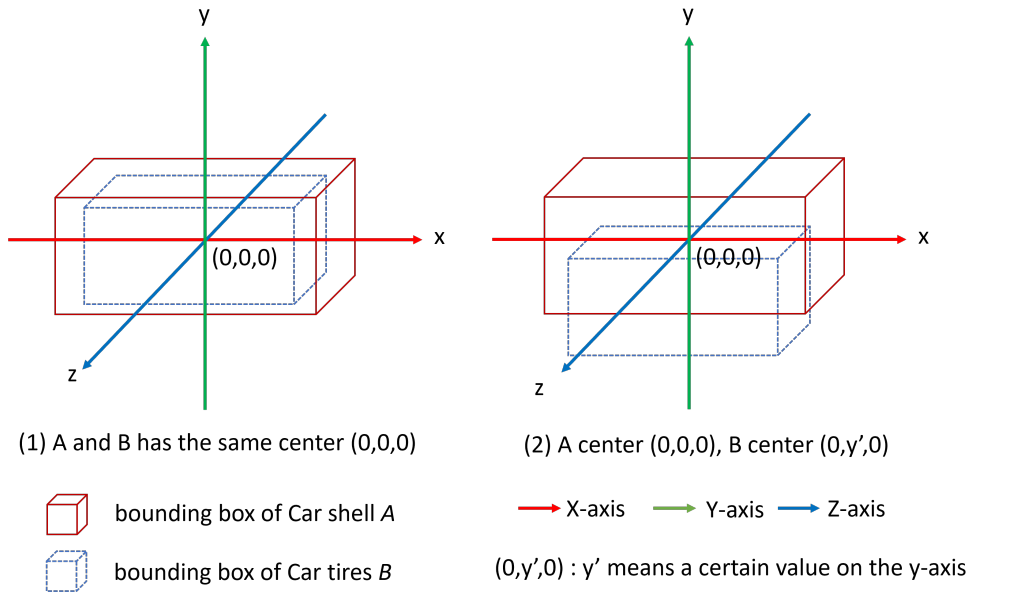
- **Center alignment.** To facilitate rapid assembly, the car shell A and the tire components B are initially aligned such that their x- and z-coordinates in (x, y, z) are set to zero. Only the y-coordinate is subsequently adjusted to resolve vertical overlaps during the assembly process. As illustrated in Figure 3.9a, the centers of the bounding boxes of both the shell and tire models are positioned at the origin $(0, 0, 0)$, ensuring that their geometric centers are aligned along the same vertical axis.
- **Maintaining proportionality.** A proportional relationship exists between the car shell A and the tire components B, with the shell generally being larger in scale. By computing and preserving a fixed proportional ratio, the size of one component can be adjusted relative to the other. Figure 3.10 illustrates this proportional relationship between the two models. Let a denote the diagonal length of the bounding box of model A, and b the corresponding diagonal for model B. The proportion between the components can therefore be represented by the ratio of these bounding-box diagonals. Note that the geometric center of the car shell's bounding box is positioned at the origin, whereas the bounding-box center of the tire is located at $(0, y', 0)$, where y' specifies the tire's translation along the y-axis.
  Specifically, the proportional relationship between the models is calculated by comparing the diagonal lengths of the bounding box of the models. The calculation of the proportion relationship is defined as follows:

$$s_r = \frac{l(b)}{l(a)} \tag{3.3}$$

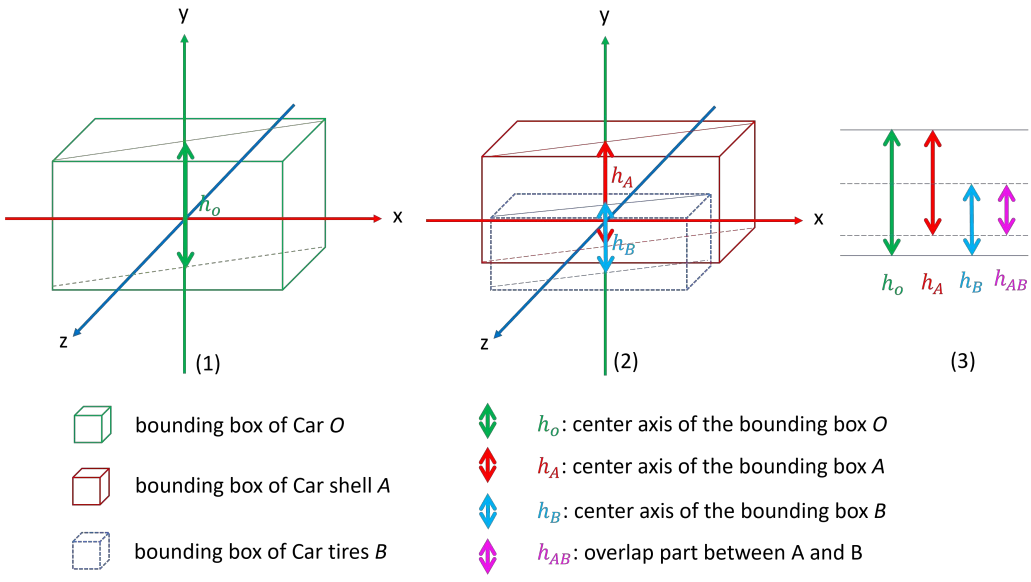
  where s_r denotes the scale ratio between the two part models. The function $l(\cdot)$ is defined to compute the diagonal length of a model's bounding box:

$$l = \sqrt{l_x^2 + l_y^2 + l_z^2} \tag{3.4}$$

  where l_x, l_y, and l_z denote the differences in the bounding-box vertex coordinates along the x-, y-, and z-axes, respectively.
- **Maintaining a fixed overlap ratio.** In the complete car model O, the bounding boxes of the car shell A and the tires B share a certain overlapping region. Preserving a fixed overlap ratio ensures that the



(a) alignment



(b) overlap

Figure 3.9: Rules used in the assembly module: (a) enforcing center alignment of component models and (b) ensuring a consistent overlap ratio between assembled parts.

parts are positioned correctly when assembled. Figure 3.9b illustrates

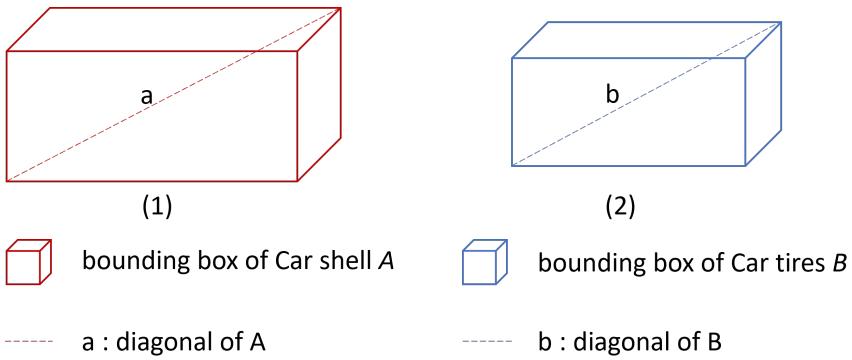


Figure 3.10: Fundamental rules used to preserve proportional relationships between component models.

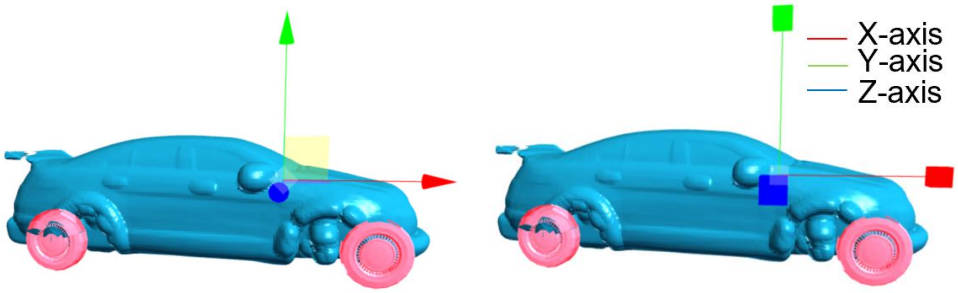
this relationship: rather than computing the full overlapping volume, the overlap can be simplified to a height-based measure due to the proportional constraints established earlier. As shown in Figure 3.9b(2), assembling the shell A and the tires B results in a fixed vertical overlap height, denoted as h_{AB}. Figure 3.9b(3) further indicates that this overlap height must maintain a consistent proportional relationship with the height of the complete model O, which is formalized in Equation 3.5.

$$o_r = h \cdot \frac{h_{AB}}{h_O} - \frac{1}{h_O} \tag{3.5}$$

where o_r indicates the overlap ratio of the overlapping height in the height of the complete model. h_{AB} denotes the height of the overlapping part between the two models, h_O represents the height of the entire overlapping area, and h represents the maximum height of the two models (i.e., the height of the tallest point in either model).

3.2.5 Model Refinement

After the assembly stage is completed, DualShape allows users to further refine the assembled model if the initial result does not meet their expectations. As illustrated in Figure 3.11, two primary editing operations are supported: adjusting the position of individual components and modifying their scale. These functions enable users to manually fine-tune the assembled parts and obtain a configuration that better matches their intended design.



(a) Translation state (b) Scaling state

Figure 3.11: Two adjustment operations in the model manipulation process: (a) translation of the tire model and (b) scaling of the tire model.

3.3 User Interface

The DualShape interface is organized into four major components: the shadow guidance module, the sketch editing module, the preview module, and the assembly module (Figure 3.12). In the shadow guidance module, the system retrieves and blends multiple contour-similar models in real time to provide users with visual references during sketching. The sketch editing module enables users to draw and modify part-level sketches through an interactive browser-based front end. The preview module displays the models that have been retrieved or generated according to the current sketch inputs. Finally, the assembly module allows users to refine the automatically assembled components by adjusting their positions and proportions.

3.3.1 Shadow Guidance

This work implements a 3D object retrieval system based on contour feature lines as input [33]. In particular, an open-source sketch search engine, OpenSSE [97], was utilized for 3D object retrieval based on sketch images as input. In addition, the function of shadow guidance was implemented based on the retrieved results.

DualShape employs 3D models as background references to support users during the sketching process. In particular, the system retrieves plausible background templates in real time based on the evolving user sketch. Whenever the user completes a stroke, the current sketch is passed to the retrieval engine, where its features are extracted and matched against those in the contour database to obtain a set of similar candidate models. These retrieved models are then blended according to a predefined transparency

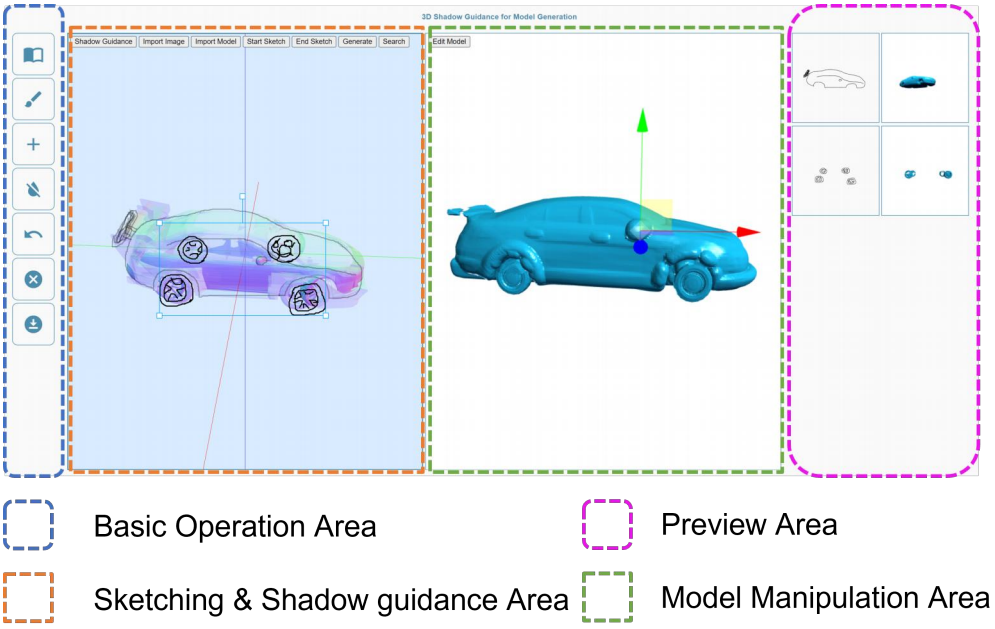


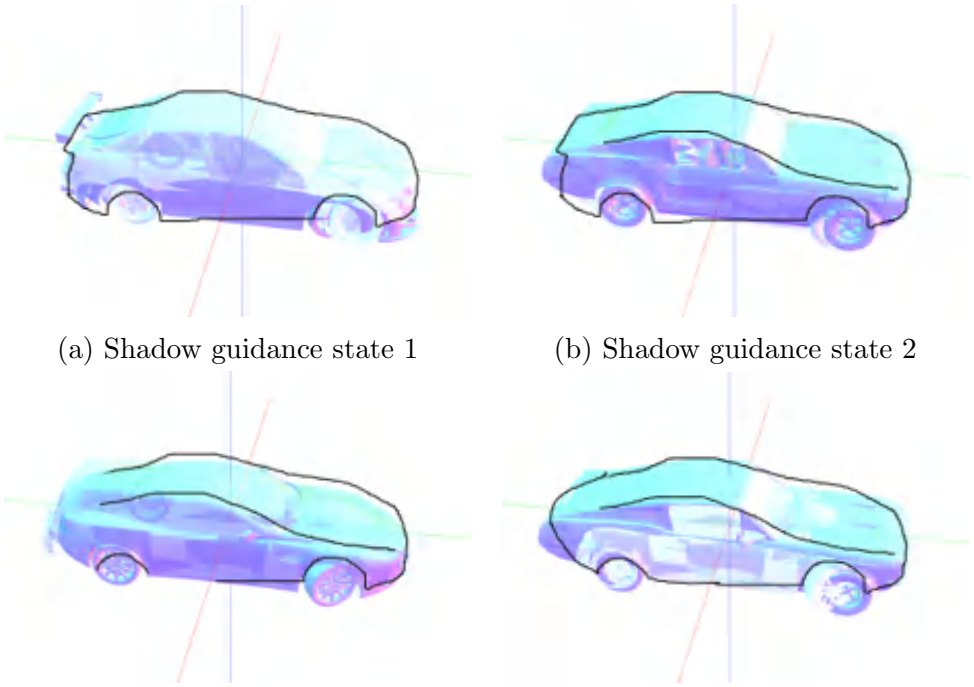
Figure 3.12: Overview of the user interface, which consists of four main regions: (a) the basic operation panel, including functions such as drawing, deleting, and downloading; (b) the drawing area, which also serves as the display region for the background model; (c) the model display and editing area, where users can enter the editing mode to adjust model details; and (d) the preview panel, showing the sketches of each part along with their corresponding 3D models.

ratio to produce a shadow-guided 3D reference model. This procedure can be expressed as follows:

$$M_{\text{output}} = \sum_{i=1}^{n} \alpha_i \cdot M_i \tag{3.6}$$

Here, M_{output} denotes the final 3D model generated after processing the user-drawn sketch. The term M_i refers to the i-th approximate model retrieved by the search engine, while α_i represents the transparency weight assigned to that model, indicating its relative contribution to the blended output. The variable n corresponds to the total number of retrieved candidate models.

The shadow-guidance mechanism allows users to explore different viewing angles of the reference model prior to sketching. Once an angle is selected, it is fixed, and DualShape transitions into the sketch-drawing phase. During sketching, the system continues to display the real-time retrieved models us-



(a) Shadow guidance state 1 (b) Shadow guidance state 2

(c) Shadow guidance state 3 (d) Shadow guidance state 4

Figure 3.13: Real-time updating of the background 3D model in response to user sketch modifications. (a) Additional strokes are drawn, prompting the shadow-guidance model to update to the state shown in (b). (b) When strokes near the base are removed, the shadow-guidance model correspondingly transitions to (c). (c) After deletion, adding new strokes again causes the guidance model to update to the configuration shown in (d).

ing this predetermined viewpoint. Figure 3.13 illustrates how the background guidance adapts dynamically as the sketch evolves.

3.3.2 Sketch Operation

The model generation process in DualShape is divided into two stages: part retrieval and part generation. Correspondingly, the input sketch is created in multiple layers, each representing the 3D structure of an individual part. After completing a sketch for one part, users may add a new drawing layer. To clearly distinguish the active layer, DualShape renders the strokes of the current layer in a darker (black) color. The sketch content from each layer is passed to the back-end generation module to produce the corresponding 3D component. Furthermore, DualShape offers both retrieval- and generation-based strategies, allowing users to obtain part models according to the

specific characteristics of each component. Figure 3.14 illustrates how the interface adapts when users sketch across different layers.

3.3.3 Preview Function

DualShape offers a real-time preview function for both sketches and generated parts, enabling users to monitor their progress and anticipate the resulting 3D model at each stage. Users may also select a specific preview layer, upon which the corresponding layer in the drawing panel is highlighted in a darker color to facilitate targeted editing. Within this view, users can perform standard operations such as panning, zooming, and adding or deleting strokes to refine the current layer.

3.3.4 Model Assembly

DualShape automatically assembles the generated or retrieved parts and presents the combined result in real time once the corresponding operations are completed. If the automatically produced assembly does not meet the user's expectations, the system provides an editing mode for further refinement. In this mode, users can select individual components and adjust them by translating or scaling along the x-, y-, or z-axis. Figure 3.11 illustrates examples of component translation and scaling during the assembly-editing process.

3.4 Results

3.4.1 Implementation Details

The real-time drawing interface of DualShape was implemented in Python on a Windows 10 platform equipped with a 3.60 GHz Intel Xeon W–2223 CPU and a GeForce RTX 3090 GPU. For the developed prototype, the average processing time for generating shadow guidance was 0.82 s, while the retrieval module required approximately 1.13 s per query. In comparison, the generation module exhibited a higher computational cost, with an average execution time of 4.23 s. This balance between fast retrieval and more intensive generation demonstrates the system's capability to efficiently manage heterogeneous tasks, ensuring a stable and responsive user experience across both retrieval-based and generation-based operations.

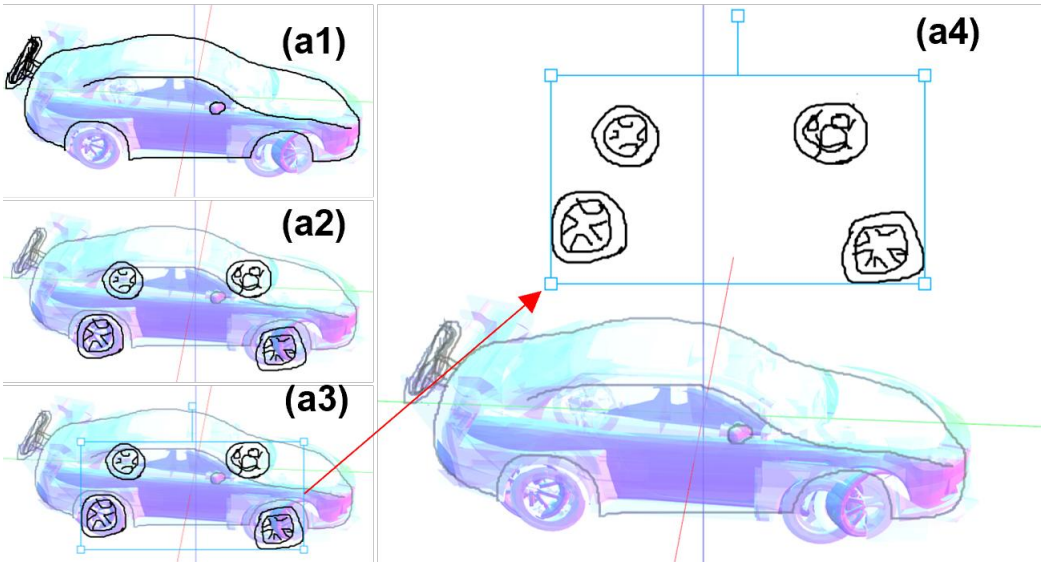
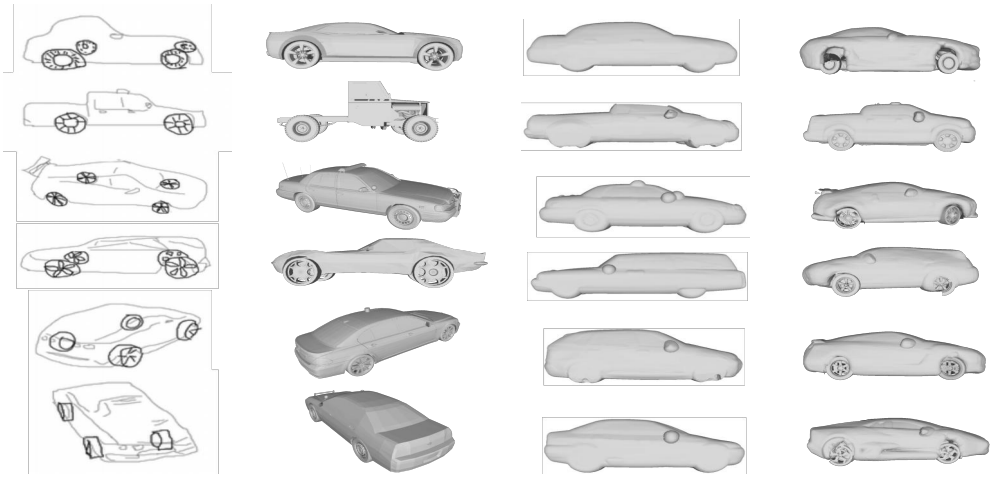


Figure 3.14: States of the sketch area under different operations. (a1) The active layer—car shell contour—is highlighted with black strokes. (a2) The second layer—tire contour—is emphasized in the same manner. (a3) When the preview function is used, selecting a layer for re-editing marks it as the active layer. (a4) In edit mode, operations such as translation can be performed; for instance, the tire sketch is moved from its position in (a3) to that in (a4).

3.4.2 Design Results

A user study was conducted with sixteen participants who were asked to design car models using the proposed system. Several examples of the user-designed models are presented in Figure 3.15. By comparing the results obtained from the same input sketches (Figure 3.15(a)) under retrieval-only and generation-only settings, it is confirmed that the hybrid strategy adopted in DualShape effectively preserves both the structural relationships between components (car shell and tires) and the fine-grained details of the tire models. In practice, participants first sketched the car shell and employed the generation module to produce multiple shell variants, thereby supporting design diversity. For the tire components, users only needed to draw the characteristic tread patterns; the retrieval module then identified suitable tire models from the dataset. Following the initial automatic assembly based on the assembly rules described in Section 3.2.4, users were able to manually fine-tune the positions and scales of individual parts to achieve their intended designs. Overall, the study demonstrated that the proposed system



(a) Sketches (b) Retrieved (c) Generated (d) Hybrid (ours)

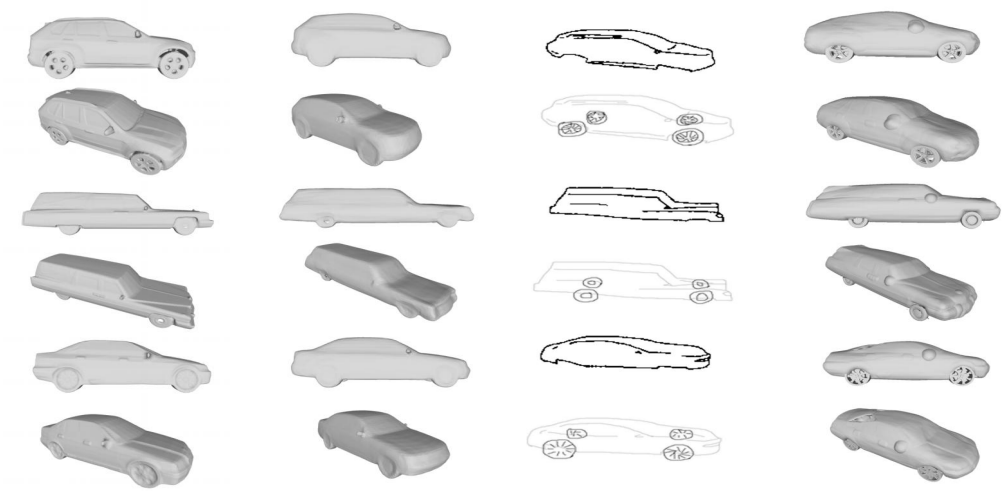
Figure 3.15: Comparison of different generation strategies. (a) User-drawn sketches of the car shell and tires; (b) retrieval-only results corresponding to the sketches; (c) generation-only outputs based on the same sketches; (d) models produced by the proposed hybrid method.

is intuitive and easy to use. DualShape significantly reduces the complexity of 3D model creation, particularly for novices with no prior experience in 3D modeling.

3.4.3 Comparison Study

To assess the effectiveness of DualShape, comparative experiments were conducted against two state-of-the-art sketch-based modeling methods: MeshSDF [57] and Sketch2Mesh [99]. MeshSDF processes the entire sketch holistically and generates a model from a global perspective, without explicitly addressing fine-grained part-level details. Consequently, the outputs produced by MeshSDF often exhibit blurred or indistinct details when different components are merged. As illustrated in Figure 3.16b, because both the car shell and the tires are generated as unified structures in MeshSDF, the connection between these components becomes unclear. Moreover, MeshSDF overlooks the local geometric characteristics of individual parts, resulting in tire models that lack distinctive and detailed features.

The models produced by DualShape are presented in Figure 3.16d. Unlike MeshSDF, DualShape treats model synthesis as the assembly of two separate components, the car shell and the tires, which enables the structural relationships between parts to be more explicitly preserved. This

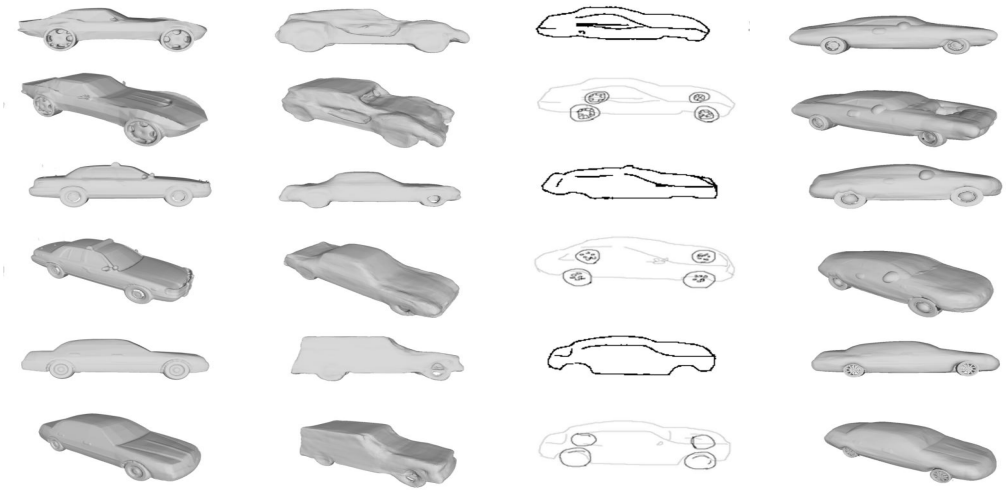


(a) Ground truth (b) MeshSDF (c) Sketches (d) Ours

Figure 3.16: Comparison with MeshSDF: (a) ground truth models; (b) models generated by MeshSDF; (c) corresponding sketch inputs; and (d) models produced by the proposed system using the same sketch inputs.

approach leads to clearer geometric interactions between components, for example the concave region on the car shell that accommodates the tire shape. Although the shell and tire models are generated independently, they can be assembled into a coherent full model without inconsistencies. As shown in Figure 3.16d, DualShape also captures fine-grained details of the tire components, producing tire models with more distinguishable and expressive features.

Sketch2Mesh employs an implicit parameterization strategy to deform and refine a 3D mesh so that its projection aligns with the external contours drawn in the sketch. However, this approach overlooks the fact that many objects, such as cars, are composed of multiple interconnected parts. As a result, it fails to account for part-level structural relationships, the geometric characteristics of component connections, and the fine-grained details present in individual parts. The car models generated by Sketch2Mesh, shown in Figure 3.17b, illustrate these limitations: the outputs lack local detail, indicating that the method primarily captures only the coarse, global structure of the vehicle. In these results, the tires and the car shell are treated as a single unified body rather than distinct components, leading to blurred or indistinct connection features. Furthermore, the generated tires exhibit only generic shapes, without representing the unique, stylized patterns typically found in real tire models; the variability between tire shapes across different models



(a) Ground truth (b) Sketch2Mesh (c) Sketches (d) Ours

Figure 3.17: Comparison with Sketch2Mesh. (a) Ground truth models; (b) models generated by Sketch2Mesh using the full sketch shown in (c); (c) input sketch provided to the system, consisting of the car shell sketch and the complete sketch including the tires; (d) model produced by our system using the sketch in (c) as input.

is minimal. In contrast, the results produced by DualShape, presented in Figure 3.17d, demonstrate clear advantages. The structural interface between the car shell and the tires is well preserved, exhibiting sharp and coherent geometric transitions. Additionally, tire components generated by DualShape show distinct stylistic differences across models, capturing the fine-grained variations ignored by Sketch2Mesh.

To evaluate the quality of the generated models, the 3D Chamfer Distance (CD-l_2) was employed as the primary metric, where lower scores indicate better reconstruction accuracy. Here, l_2 denotes the Euclidean norm, which measures the discrepancy between two point clouds by computing the squared distances between corresponding points. The metric is obtained by uniformly sampling $N = 20{,}000$ points from the reconstructed mesh to form the first point cloud $\mathbf{C}_1$, and sampling the same number of points from the ground truth mesh to construct the second point cloud $\mathbf{C}_2$. The CD-l_2 is defined as follows:

$$\text{CD-}l_2 = \frac{1}{N}\sum_{x\in\mathbf{C}_1}\min_{y\in\mathbf{C}_2}\|x-y\|^2 + \frac{1}{N}\sum_{y\in\mathbf{C}_2}\min_{x\in\mathbf{C}_1}\|y-x\|^2 \tag{3.7}$$

In addition, a normal consistency (NC) metric was employed to assess the

Table 3.1: Evaluation metrics of model quality.

Method	Metric	
	$CD\text{-}l_2 \cdot 10^3 \downarrow$	$NC \cdot 10^2 \uparrow$
MeshSDF	4.28	90.68
Sketch2Mesh	3.09	90.75
Ours	2.53	89.21

alignment of surface normals, with higher values indicating better geometric consistency. The NC score is computed as the average absolute dot product between the normals of the reconstructed mesh G and the normals of the nearest corresponding points on the ground truth mesh R. It is formally defined as follows:

$$NC(G, R) = \frac{1}{|R|} \sum_{r \in R, g \in G} |r \cdot g| + \frac{1}{|G|} \sum_{r \in R, g \in G} |g \cdot r| \tag{3.8}$$

Here, G and R denote the sets of surface normals from the reconstructed mesh and the ground truth mesh, respectively. The terms $|G|$ and $|R|$ represent the number of normals in each mesh. The metric computes the average absolute dot product between corresponding normals, thereby quantifying the degree of normal alignment between the reconstructed result and the reference mesh.

As summarized in Table 3.1, the proposed method achieves a CD-$l_2 \cdot 10^3$ score of 2.53, outperforming both MeshSDF (4.28) and Sketch2Mesh (3.09). A lower Chamfer Distance reflects a closer geometric match to the ground truth, indicating that the method accurately captures the object's spatial structure. In terms of normal consistency, the proposed approach attains an NC $\cdot 10^2$ value of 89.21, which is comparable to that of MeshSDF (90.68) and Sketch2Mesh (90.75). This demonstrates that the reconstructed surfaces preserve a high level of normal agreement with the ground truth.

3.5 User Study

A user study was conducted to evaluate the effectiveness of the DualShape user interface. Sixteen participants were recruited for the experiment, consisting of ten male and six female graduate students. Each participant used the system to design car models and subsequently completed a questionnaire. The collected responses were analyzed statistically to assess

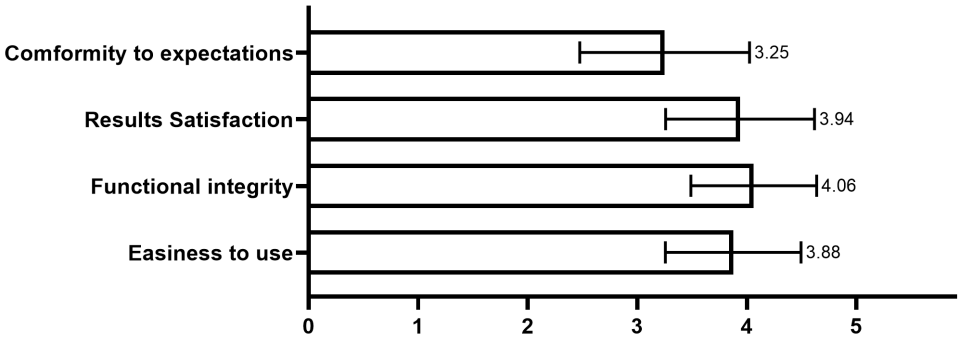


Figure 3.18: Overall evaluation results.

users' perceptions of the interface. The questionnaire consisted of three components: an overall evaluation of the system, an assessment based on the System Usability Scale (SUS), and an evaluation of specific interface functionalities.

3.5.1 Overall Evaluation

After completing the experiment, participants were asked to rate four aspects of the system, "system functional integrity," "user interface convenience," "satisfaction with generated results," and "conformity to expectations", using a 5-point Likert scale (1 = strongly disagree, 5 = strongly agree). As shown in Figure 3.18, the distributions of the responses were analyzed for each indicator. Most participants rated the system's functional integrity at level 4, indicating a generally positive evaluation. Ratings for user interface convenience were primarily within the range of 3 to 4. Satisfaction with the generated results was also concentrated at level 4, and the majority of users agreed that the generated outputs met their expectations.

The mean rating for the ease of use of the interface was 3.88, with a standard deviation of 0.62, indicating that most participants found the interface intuitive to operate. The perceived functional completeness of the system received a mean score of 4.06 (standard deviation = 0.57), suggesting that users regarded the available functionalities as sufficiently comprehensive. Satisfaction with the generated models yielded a mean rating of 3.94 and a standard deviation of 0.68, demonstrating that the majority of users were pleased with the output quality. Regarding the extent to which the generated models met user expectations, the mean score was 3.25, with a standard deviation of 0.77, reflecting general agreement that the results were aligned

Table 3.2: Results of the post-experiment SUS metrics questionnaire. ⇑ indicates that higher scores are better; ⇓ for the other case. The total score is 81.67 out of 100.

	Questions	**Mean**	**SD**
1	I would like to use this system frequently. ⇑	3.75	0.68
2	I found this system unnecessarily complex. ⇓	1.44	0.51
3	This system was easy to use. ⇑	3.88	0.62
4	I would need the support of a technical person to be able to use this system. ⇓	1.75	0.45
5	I found the various functions in this system were well integrated. ⇑	4.06	0.57
6	I thought there was too much inconsistency in this system. ⇓	1.81	0.40
7	I would imagine that most people would learn to use this system very quickly. ⇑	4.19	0.40
8	I found this system very cumbersome to use. ⇓	1.31	0.48
9	I felt very confident in using this system. ⇑	3.88	0.50
10	I needed to learn a lot of things before I could get going with this system. ⇓	1.06	0.25

with users' expectations.

3.5.2 Subjective Evaluation

In addition to the overall evaluation, participants completed a System Usability Scale (SUS) questionnaire to further assess the usability of DualShape. Table 3.2 summarizes the SUS evaluation results. All participants reported that the interface was generally satisfying, and most agreed that new users would be able to learn the system quickly. Furthermore, 75% of the participants indicated that they would be willing to use the system frequently for designing car models, and 77.6% agreed that the interface functions were easy to operate. In addition, 81.2% of the participants considered the system's functionalities to be well integrated, and 77.6% expressed confidence while using the system. Overall, DualShape achieved a SUS score of 80.94 out of 100, indicating excellent usability of the interface.

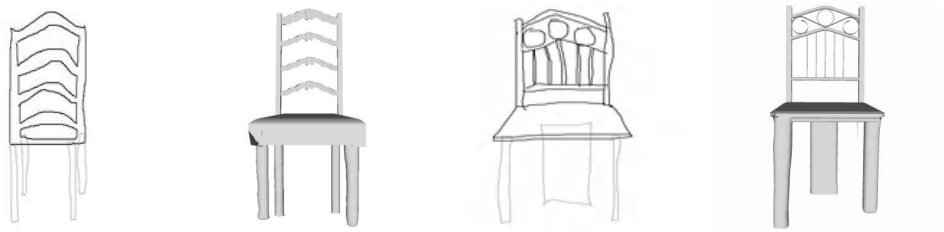
Table 3.3: Results of specific functions evaluation questionnaires.

	Questions	**Mean**	**SD**
1	Sketching in layers is in line with the drawing habit.	3.56	0.96
2	The shadow guide function can adequately assist me in sketching.	3.75	0.58
3	Shadow guidance is easy to use.	4.06	0.44
4	The generate function was useful for generating car shell models.	4.00	0.63
5	The car shells generated using the generation method meet expectations.	3.44	0.51
6	It was useful to use the search function to retrieve the models.	4.25	0.45
7	The model retrieved using the search function meets expectations.	4.50	0.52
8	The drawing function is easy to use.	4.31	0.48
9	The delete function is easy to use.	2.38	0.50
10	The undo function is easy to use.	4.44	0.53
11	The download function is easy to use.	4.75	0.58
12	The preview function provided can be a good way for me to observe the model generation process.	4.12	0.72
13	The function of re-editing after selecting the preview layer is very useful.	4.06	0.44
14	The assembly function is useful.	4.31	0.48
15	The re-edit function is useful.	4.69	0.48
16	The re-edit function is easy to use.	3.63	0.50

3.5.3 Specific Functions Evaluation

To further assess the usefulness and convenience of the designed interface functionalities, participants were also asked to evaluate the specific features they interacted with during the experiment. The results of this functional evaluation are summarized in Table 3.3.

The layer-by-layer part-sketching workflow adopted in DualShape was well received by participants; 71.2% reported that this approach aligned with their natural drawing habits and enabled faster design. Furthermore, 75% of participants indicated that the 3D shadow-guidance feature was helpful for



(a) Input sketch1 (b) Chair model1 (c) Input sketch2 (d) Chair model2

Figure 3.19: Examples of designed chair models. (a) and (c) show the user-drawn chair sketches used as input; (b) and (d) present the corresponding models generated by the proposed hybrid method.

quickly outlining the car shape, and 81.2% agreed that the shadow-guidance mechanism was intuitive, providing real-time references that closely matched their sketches.

In addition, 80% of the participants agreed that using the generation module for producing car shells was appropriate, and 68.8% felt that the generated shell models generally met their expectations. Regarding the tire component, 85% of the participants recognized the usefulness of the retrieval-based method, and 90% expressed satisfaction with the retrieved tire models. For the basic interface operations—drawing, deleting, undoing, and downloading— 86.2%, 47.6%, 88.8%, and 95% of the participants, respectively, considered these functions to be user-friendly. With respect to the preview and re-edit sketch features, 82.4% indicated that the preview function provided a clear understanding of the model-generation process, and 81.2% acknowledged the necessity of the re-edit function. However, only 55% reported satisfaction with the re-editing capabilities for modifying sketches.

Regarding the real-time display of the assembled model, 86.2% of participants responded positively. Furthermore, 93.8% acknowledged that the ability to adjust the details of the assembled model was both necessary and useful, and 72.6% indicated that the model-adjustment functionality was user-friendly.

3.6 Discussion

This section discusses the feasibility of the proposed shape-design approach, along with its limitations and potential directions for future work. The discussion first examines the scalability of the system and the diversity of models that can be created through the hybrid design framework. Subsequently, the limitations of the current method are outlined and several failure cases are

presented to illustrate where the system does not perform as expected. Based on these observations, possible improvements and future research directions are proposed to address these limitations.

3.6.1 Shape Design

The proposed framework can be applied to a wide range of shape-design tasks, and its extensibility enables adaptation to multiple model categories. As a simple demonstration, chair design was explored as an extension of the system. Examples of user-created chair models are shown in Figure 3.19. For this task, the chair was divided into two components: the body (backrest and seat) and the legs. Users sketched the characteristic outline of the chair body and retrieved the most similar component from the dataset, while the leg component was produced using the generation module based on the user's sketch. These results demonstrate that the framework can be readily extended to other object categories, indicating that DualShape possesses strong scalability and adaptability for diverse model-design scenarios.

3.6.2 Relation to Sketch-Based Architectural Modeling

Recent studies show that sketch-based techniques are also widely applied in architectural design. A literature review by Li et al. [47] indicates that generative AI has become an important tool for early architectural exploration, where sketches function as concise inputs for expressing spatial and structural intent. As illustrated in Figure 3.20, Sketch-to-Architecture [100] further demonstrates that free-hand building sketches can be converted into floorplans, massing models, architectural renderings, and 3D models.

Although such architectural systems operate at a larger spatial scale, they share the same core principle as the method proposed in this chapter: sketches serve as structural constraints for organizing geometric generation. This conceptual link naturally leads to the next chapter, which extends the sketch-to-structure idea to automatic multi-Level-of-Detail (LoD) sketch construction for architectural modeling.

3.6.3 Limitations and Future Work

One limitation of the proposed approach is that the initial automatic assembly of different parts currently relies on manually defined prior knowledge specific to each object category. As demonstrated in the chair-design example, no dedicated assembly rules were established for positioning and scaling the chair components, and users were required to manually adjust the

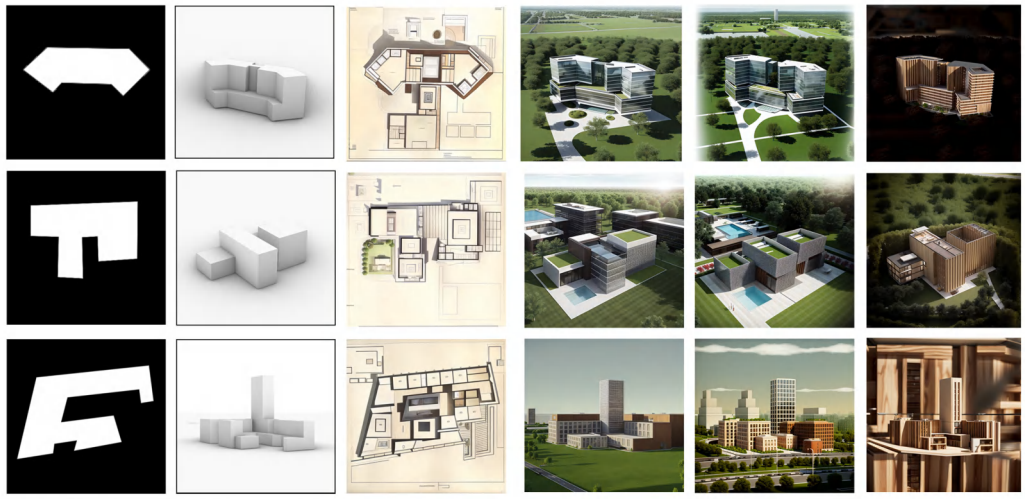
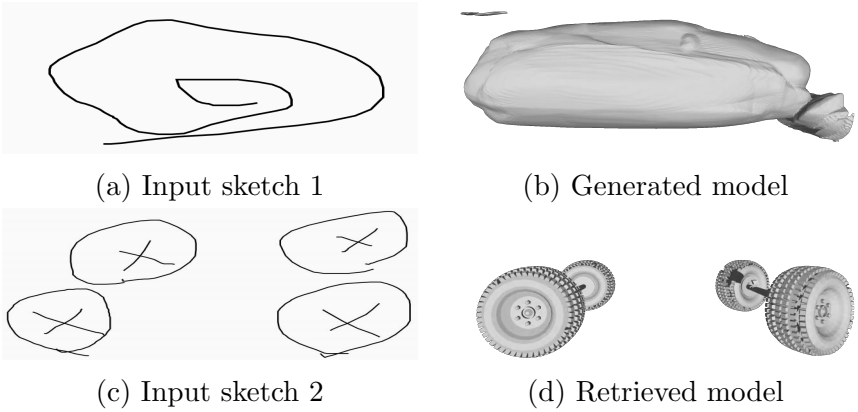


Figure 3.20: Examples of generating floorplans, massing models, architectural renderings, and 3D models from sketches [100].

relative placement and proportions of the parts (Figure 3.19). To address this limitation, a promising direction for future work is to incorporate a learning-based module that can automatically infer the positional and proportional relationships between component shapes. Such a module would eliminate the need to handcraft category-specific assembly rules and would enable fully automated part assembly across different object classes.

Another limitation arises when the input sketch is overly simple, which can lead to suboptimal results in both part generation and retrieval. Several examples of such failures are shown in Figure 3.21. When the sketch provides only sparse or insufficient structural cues, the retrieval module may return models that do not align with the user's intended design. To mitigate this issue, a promising direction for future improvement is to incorporate a contour-optimization module that enriches the structural information of the sketch. Such an enhancement would not only produce sketches with clearer geometric features but also improve the quality and reliability of the retrieval results.

In addition, the current system is limited to a single object category—car models, which restricts its general applicability. A natural extension of this work is to broaden the range of supported object classes, including everyday items such as airplanes, tables, and vases, as well as more structurally complex categories such as animals and human figures. Expanding the dataset with more diverse and informative examples would further enhance



(a) Input sketch 1 (b) Generated model

(c) Input sketch 2 (d) Retrieved model

Figure 3.21: Failure cases in part generation and retrieval. (a) and (c) show user-drawn input sketches; (b) illustrates the failed result produced by the generation module from sketch (a); (d) shows the failed retrieval result obtained from sketch (c).

the quality of the generated results. Finally, variations in the resolution of models produced by the hybrid approach may still occur; this issue could be addressed through mesh-optimization techniques in future improvements.

3.7 Conclusion

This chapter introduces DualShape, a hybrid 3D shape–design framework based on part assembly, which integrates both retrieval-based and generation-based strategies to obtain component models and subsequently compose them into a complete object. Furthermore, a shadow-guided drawing mechanism was incorporated, using 3D models as visual references to assist users during sketching. Built upon this framework, an interactive user interface was developed and implemented, enabling users to generate models from sketches while also leveraging retrieved parts to preserve fine-grained geometric details. In addition to comparative experiments with existing methods, a user study was conducted to evaluate the effectiveness, usability, and overall practicality of the proposed framework and interface.

Building on these object-level mechanisms, the next chapter extends sketch-guided structural generation to the architectural level. While DualShape shows how free-hand sketches can support part-aware 3D generation for individual objects, architectural design additionally requires hierarchical abstraction and consistent multi-Level-of-Detail (LoD) representations. Chapter 4 therefore introduces an automatic LoD sketch construction frame-

work that derives aligned contour, massing, depth, and semantic layers from detailed building models. This transition establishes the multi-LoD data foundation for subsequent building-level generative modeling and supports the multi-view generation and facade renovation studies presented in Chapters 5 and 6.

Chapter 4

LoD Sketch Construction for Generative Architectural Modeling

This chapter establishes the foundation of the dissertation's building-level generative modeling framework by focusing on buildings as a specific class of objects and constructing geometrically aligned multi-Level-of-Detail (LoD) sketch representations [23] (Figure 4.1). As an intermediate representation between conceptual design intent and data-driven generation, LoD-consistent sketches provide the structural cues required for controllable architectural modeling across different levels of semantic and geometric abstraction.

Building on the object-level structural reasoning developed in the previous chapter, this study addresses a major bottleneck in architectural artificial intelligence, namely the lack of high-quality and hierarchically aligned LoD training data. Traditional LoD modeling workflows depend heavily on manual procedures, which are time-consuming and prone to geometric inconsistency. To overcome these limitations, this chapter proposes an automatic LoD sketch extraction framework that integrates computer vision techniques with generative AI methods. The framework progressively transforms detailed architectural models into aligned contour, massing, depth, and semantic representations, formalizing sketches as a multi-level computational hierarchy suitable for data-driven generative modeling.

The proposed extraction pipeline exhibits stable geometric performance across LoD transitions. Experimental evaluations report structural similarity values of 0.7319 from LoD3 to LoD2 and 0.7532 from LoD2 to LoD1. The corresponding normalized Hausdorff distances are 25.1 percent and 61.0 percent of the image diagonal. These results indicate that the framework preserves global volumetric structure while achieving controlled semantic simplification, producing representations that remain consistent and computationally usable across different LoD levels.

By providing reliable and geometrically aligned LoD representations, this chapter establishes the data foundation and semantic alignment mechanisms for the multi-view architectural generation and facade renovation studies presented in Chapters 5 and 6. The resulting LoD dataset enables diffusion-

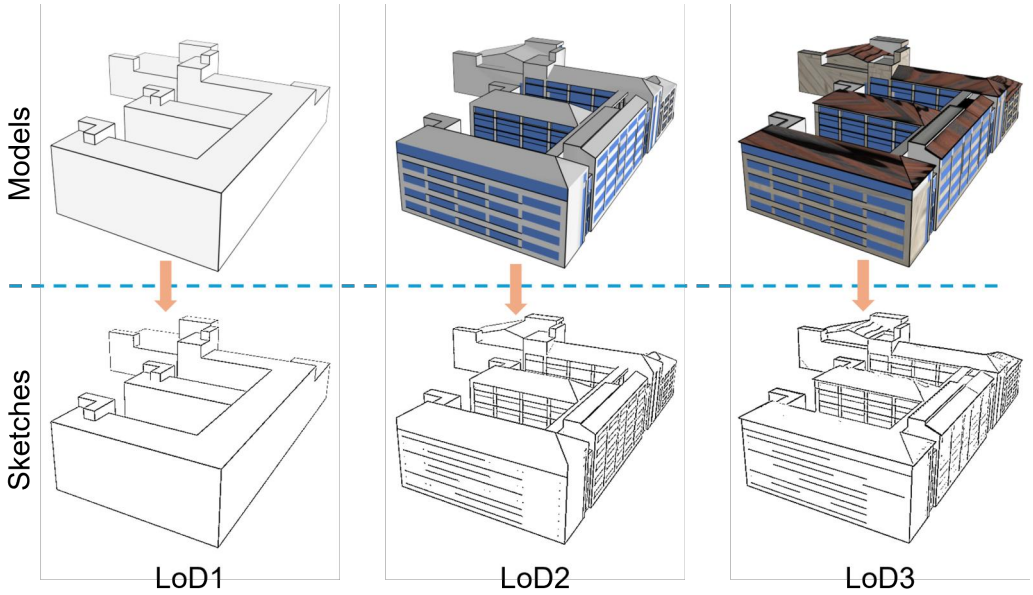


Figure 4.1: Overview of LoD1–LoD3 architectural models (top) and their corresponding sketch abstractions (bottom).

based generative models to learn the progressive transition from conceptual sketches to semantically enriched architectural forms, forming a central component of the dissertation's building-level generative framework.

4.1 Background

In architectural design practice, models are represented at different Levels of Detail to support the transition from conceptual massing to detailed design. According to the CityGML standard [101], LoD1 represents volumetric blocks, LoD2 incorporates simplified roof geometry and semantic components, and LoD3 includes detailed facade elements such as windows and doors. Despite these definitions, constructing and maintaining multi-LoD models in BIM and digital twin systems heavily depends on manual operations. These processes are time consuming and susceptible to geometric inconsistency across levels.

LoD concepts have been examined extensively in Building Information Modeling and urban digital twin research. Earlier studies outlined standardized hierarchies that span from coarse massing to detailed facade representations [59, 102]. Although these hierarchies define a common representational structure, most automated approaches focus on refining low-LoD models

into higher-LoD forms. Downward simplification from LoD3 to lower LoD levels has received limited attention and the absence of well-aligned multi-LoD datasets continues to restrict progress in automated simplification and hierarchical model control [103, 104].

Sketch extraction research provides additional insight for automated LoD construction. Sketches serve as essential representational media in architectural design and help connect conceptual reasoning with spatial configuration. Classical image-based extraction methods such as Canny, Sobel and Laplacian filters can identify contour information but are highly sensitive to illumination and texture variations. Deep learning-based approaches such as HED and RCF [27, 28] support multi-scale and semantically coherent line extraction. Learning-based sketch extraction for architectural and heritage imagery has also been proposed and these techniques have improved geometric consistency [105]. However, obtaining structural abstraction that remains consistent across multiple levels of detail continues to present challenges for sketch based architectural modeling.

Generative AI has further expanded possibilities in architectural design. Diffusion-based generative models and conditional control frameworks such as Stable Diffusion [7], ControlNet [8] and T2I-Adapter [70] can synthesize high-fidelity and semantically coherent images using inputs such as sketches, text or depth. These methods have supported research on sketch-to-facade generation and multi-stage LoD representation learning [47, 106, 107]. The performance of such models depends on paired training data that capture structural relationships across LoD transitions. Existing architectural datasets commonly contain a single LoD level and lack paired samples that reflect the evolution of geometric detail. The absence of aligned multi-LoD data limits the ability of generative models to learn reliable structural abstraction and controllable transformation.

The automatic LoD sketch extraction method introduced in this chapter aims to address these challenges by integrating computer vision techniques with generative modeling. The method constructs mappings from high-LoD models to simplified sketch-based LoD representations and generates a paired multi-LoD dataset that supports data-driven generative architectural modeling. As illustrated in Figure 4.2, the extraction process creates sketches with controlled levels of abstraction and consistent geometric characteristics across transitions. Through this process, a standardized multi-LoD sketch dataset is established that can support future research in data-driven architectural generation.

The main contributions of this work include:

- An automated LoD sketch extraction framework that achieves geomet-

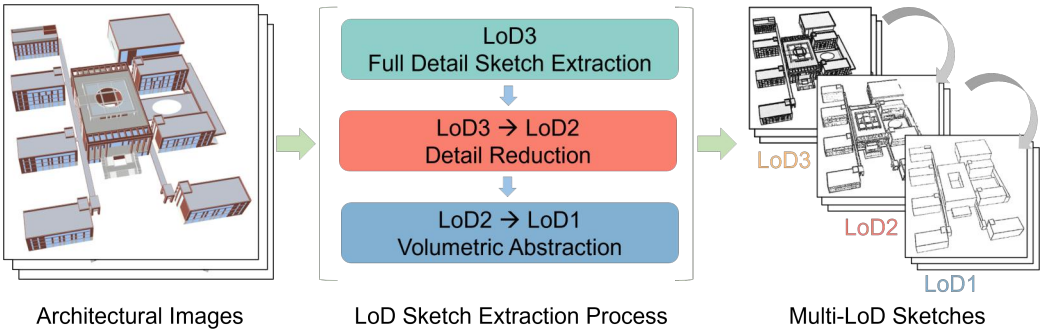


Figure 4.2: Overview of the proposed Automatic LoD Sketch Extraction Framework.

rically consistent transformation from high-detail models to multi-level sketches;

- A paired multi-LoD sketch dataset, offering standardized training samples for generative AI-based multi-stage architectural modeling.

4.2 Methodology

An automatic LoD sketch extraction framework (Figure 4.2) based on generative AI is proposed, aiming to automatically generate geometrically consistent and detail-controllable multi-level sketches from high-detail architectural models.

4.2.1 Full-Detail Sketch Extraction

At the first stage (LoD3), the goal is to automatically generate high-fidelity line sketches from architectural renderings that contain complete texture and lighting information, serving as the starting point for the multi-level architectural simplification process. The objective at this stage is to preserve structural and detailed edges of buildings while suppressing shadows and noise, ensuring that the resulting sketches exhibit uniform line thickness and clear contour hierarchy.

To achieve this, a full-detail sketch extraction pipeline is designed based on image gradient analysis and morphological enhancement. Figure 4.3 shows the intermediate results of the full-detail sketch extraction process at the LoD3 stage.

First, the input color rendering is converted into a grayscale image G, and the Sobel operator is applied to compute horizontal and vertical gradient

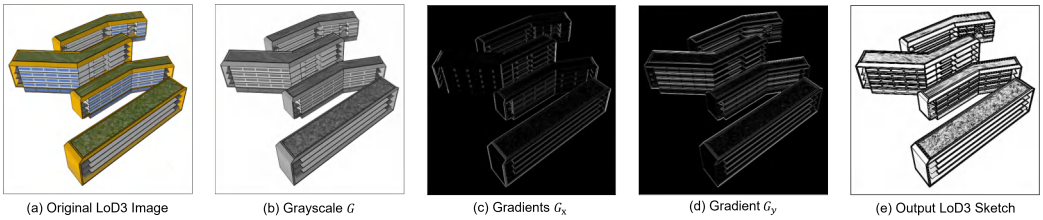


Figure 4.3: Intermediate results of the full-detail sketch extraction process at the LoD3 stage.

components G_x and G_y, respectively. The gradient magnitude M is then calculated as:

$$M = \sqrt{G_x^2 + G_y^2}, \tag{4.1}$$

which is followed by an inversion operation $(255 - M)$ to obtain an edge brightness distribution consistent with human visual perception, resulting in an edge map E.

Next, two adjustable parameters are introduced, namely the shadow blending coefficient α and the line thickness coefficient β, which control the retention of grayscale information and the enhancement of line intensity, respectively. The overall transformation is defined as:

$$S = (1 - \alpha)E + \alpha G, \tag{4.2}$$

$$O = (S - 128) \times (1 + 2\beta) + 128, \tag{4.3}$$

where S represents the intermediate blended result and O denotes the final sketch output. The constant 128 serves as the mid-gray reference point, ensuring that global brightness remains stable during contrast enhancement. A smaller α produces a purer line-drawing appearance, while a larger α preserves more shading information. Conversely, β controls the strength and thickness of the lines—higher values produce darker and bolder contours, whereas lower values yield finer, more delicate strokes.

Finally, morphological Black-Hat enhancement is applied to darken fine-line regions, improving line contrast and visual uniformity without affecting broader grayscale areas. This process ensures that the generated LoD3 sketches maintain both structural fidelity and visual clarity, providing a reliable foundation for subsequent LoD simplification stages.

4.2.2 Detail Reduction Using Generative Modeling

The second stage (from LoD3 to LoD2) aims to reduce local components and texture details in architectural models, enabling a transition from detailed

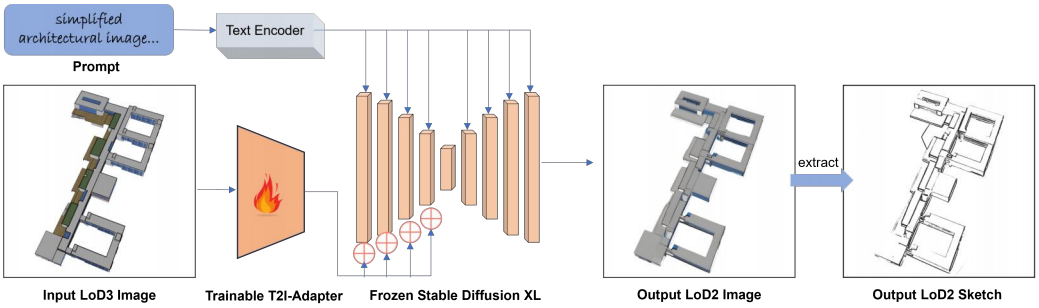


Figure 4.4: Detail reduction from LoD3 to LoD2 using a generative modeling pipeline.

and feature-rich representations to simplified structural forms. However, directly training a model using paired LoD3 and LoD2 sketches often fails to produce meaningful results. After sketch extraction, the two levels exhibit highly similar edge distributions, where texture and structural lines become intertwined, leaving only subtle geometric differences. As a result, the model struggles to discern which lines should be preserved or removed, leading to unstable convergence and inconsistent detail reduction. Moreover, sketches inherently contain only edge information while lacking mid-level semantic cues such as lighting, materials, and shading, which are factors essential for understanding the concept of detailed level.

To address these challenges, a two-stage approach is adopted. Specifically, LoD3 images are first converted into LoD2-style RGB images, allowing the model to learn the "fine-to-coarse" transformation in the image domain, where richer visual features are available. In the second stage, sketches are extracted from the generated LoD2 images, producing structurally clear and detail-simplified LoD2 sketches. This indirect image-based approach provides stronger learning signals and better geometric consistency in the generated results.

As illustrated in Figure 4.4, during the first stage a Stable Diffusion XL (SDXL) architecture [108] integrated with a T2I Adapter [70] module is employed to generate LoD2 style images from LoD3 inputs. The T2I Adapter is adopted instead of the ControlNet [8] architecture because its mechanism of feature modulation and injection is better suited for flexible detail abstraction. ControlNet imposes strong spatial constraints on the generation process and often forces the model to follow the fine grained edge structures of the LoD3 input strictly. In contrast, the T2I Adapter provides a lighter and more adaptive conditioning pathway that allows the model to learn a soft mapping from complex visual features to simplified ones. This

flexibility enables the network to suppress facade textures and window details effectively while preserving global proportions and spatial coherence.

During training, the LoD3 image serves as the conditional input, and the corresponding LoD2 image is used as the generation target. Short textual prompts (e.g., "LoD2 style, simplified structure, no small fixtures or textures") are provided for lightweight semantic guidance. Through this process, the model learns to perform a visual mapping from complex to simplified structural representations, generating RGB images that exhibit LoD2-level abstraction.

Next, the generated LoD2 image is transformed into its sketch representation using the same extraction process described in Section 4.2.1. The generated LoD2 sketches retain consistent building mass boundaries, roof outlines, and floor separations, while effectively removing fine-grained details such as window frames, surface textures, and decorative patterns.

4.2.3 Volumetric Abstraction with ControlNet

The third stage (from LoD2 to LoD1) aims to further abstract the LoD2 sketches into LoD1 representations that preserve only the primary building volumes. Since LoD2 and LoD1 differ significantly in both semantic hierarchy and geometric structure, using a single conditional input often fails to achieve a balance between shape preservation and spatial abstraction. To address this issue, a dual-ControlNet abstraction framework is designed, which jointly leverages structural sketches and depth information to learn the mapping from detailed structures to simplified volumetric forms.

During training, the model takes the LoD2 sketch and its corresponding depth map as dual conditioning inputs, which are processed by two specialized branches. The Sketch ControlNet preserves the global contour and geometric proportions of the building, preventing structural distortion or misalignment during abstraction. The Depth ControlNet provides constraints on spatial hierarchy and volumetric relationships, enabling the model to remove small components such as doors and windows while maintaining correct depth ordering between building masses.

The multi-scale features extracted from both Sketch and Depth ControlNets are injected into the main UNet backbone at downsampling and intermediate layers, where they modulate the latent diffusion process together with the noise embeddings. This mechanism allows the model to achieve a progressive transition from detailed structures to abstract volumes during denoising. Training is guided by the standard noise-consistency loss of

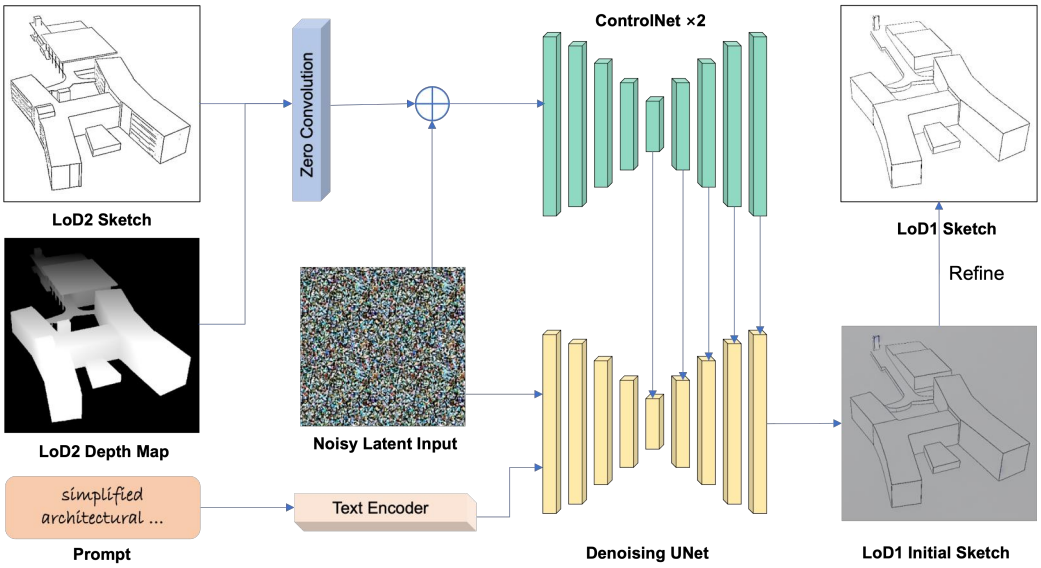


Figure 4.5: Volumetric abstraction from LoD2 to LoD1 using a dual-ControlNet framework.

diffusion, minimizing the difference between the predicted and true noise:

$$\mathcal{L}_{\text{train}} = \mathbb{E}_{t,\epsilon} \left\| \epsilon_\theta(x_t, t, c_{\text{sketch}}, c_{\text{depth}}) - \epsilon \right\|^2, \tag{4.4}$$

where c_{sketch} and c_{depth} denote the sketch and depth conditioning inputs, respectively. The model is trained end-to-end on LoD2–LoD1 pairs to automatically remove small-scale features such as openings, ornaments, and facade details while preserving structural integrity.

During inference, the system requires only a LoD2 sketch to generate the LoD1 abstraction. For scenes without available depth annotations, the MiDaS [109] pretrained model is employed to estimate a depth map directly from the input LoD2 sketch. This estimated depth is then fed into the Depth ControlNet as auxiliary guidance, together with the sketch condition, to support volumetric abstraction generation. Throughout the multi-step diffusion sampling process, the model progressively removes fine details while retaining the main building volumes and spatial hierarchy. Finally, the generated LoD1 image is post-processed using Canny edge detection and line refinement to obtain a geometrically consistent and hierarchically clear LoD1 sketch.

4.3 Experiments and Results

4.3.1 Experimental Setup and Data Preparation

A total of 50 groups of architectural models were constructed, including LoD1, LoD2, and LoD3 representations of 150 models. Each model was rendered using *Blender* and a *pyrender*-based orbit-capture script with 36 azimuth angles (0°–350°, step 10°) and 7 elevation angles (0°–60°), producing 252 viewpoints per LoD. For each viewpoint, both RGB and depth images were rendered at a resolution of 512×512 pixels, resulting in 504 images per model. Across all LoD levels and 50 model groups, the dataset contained approximately 75,600 images (37,800 RGB and 37,800 depth). All data were standardized and geometrically aligned to maintain consistency across LoD levels. This dataset provides comprehensive paired samples for the multi-stage generation framework, which subsequently supports the experiments on detail extraction, structural reduction, and volumetric abstraction. All experiments were conducted on a workstation equipped with a GeForce RTX 5090 GPU. The experimental framework consists of three main stages, corresponding to different LoD transitions. All experiments were performed under fixed random seeds and unified rendering settings to ensure reproducibility.

4.3.2 Qualitative Results

The proposed framework demonstrates strong capability in generating hierarchical sketch representations that progressively abstract architectural structures across LoD levels. As illustrated in Figure 4.6, the process begins with LoD3 images, from which LoD3 sketches are extracted to capture the primary structural lines and compositional organization of the buildings. Subsequently, through the generative transformation from LoD3 to LoD2, the model removes secondary textures and decorative details while preserving the essential geometric boundaries, resulting in sketches that emphasize clean structural outlines and spatial coherence.

Finally, the LoD1 sketches produced by the dual-ControlNet abstraction stage depict simplified, shoebox-like volumes that clearly express the core building masses and their spatial hierarchy. Across all levels, the generated sketches exhibit high geometric consistency and accurate proportion alignment, ensuring smooth transitions between abstraction stages.

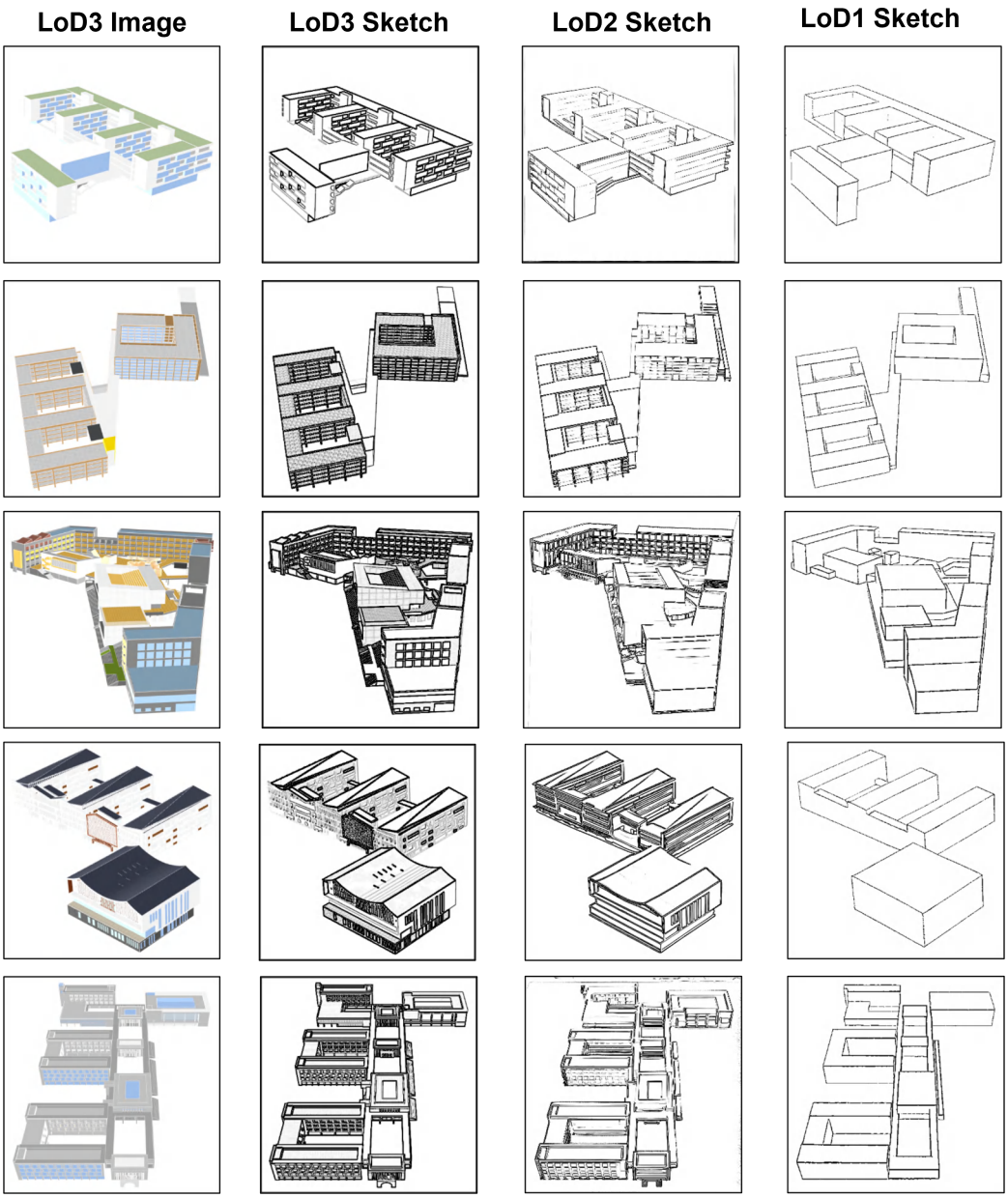


Figure 4.6: Qualitative results of the proposed framework showing progressive abstraction from LoD3 image to LoD1 sketch.

4.3.3 Quantitative Evaluation

The quantitative comparison shows that the metric distributions of the two stages align precisely with their respective functional objectives, confirming the rationality and effectiveness of the proposed hierarchical simplification

Table 4.1: Quantitative comparison of two LoD transition stages.

Stage	**SSIM (↑)**	**MSE (↓)**	**HD (px) (↓)**	**Normalized HD (% diag.) (↓)**
LoD3 → LoD2	0.7319	5.15×10^3	181.90	25.1%
LoD2 → LoD1	0.7532	4.24×10^3	441.65	61.0%

framework. All images are standardized to a resolution of 512 × 512, corresponding to a diagonal length of approximately 724.1 pixels, which allows the Hausdorff Distance (HD) to be normalized for scale consistency across stages.

As summarized in Table 4.1, during the LoD3-to-LoD2 stage, the model achieves an SSIM of 0.7319, an MSE of 5.15×10^3, and a normalized Hausdorff Distance equal to 25.1% of the image diagonal. These results demonstrate that the proposed generative simplification effectively eliminates local textures and redundant line segments while preserving the overall geometric configuration of the building. The MSE accounts for only 7.9% of the maximum possible pixel error (65,025 for 8-bit images), indicating that most changes occur at small-scale structural features rather than in the global geometry. This trend substantiates the intended role of this stage—detail reduction while maintaining geometric fidelity.

In contrast, during the LoD2 to LoD1 stage, the SSIM slightly increases to 0.7532 and the MSE further decreases to 4240 (approximately 6.5% of the 8 bit range) which reflects globally cleaner and more homogeneous sketch representations. However, the HD rises sharply to 441.65 pixels (61.0% of the image diagonal). This increase should not be interpreted as degradation. Instead it reflects the intentional geometric abstraction inherent to volumetric simplification, where the model deliberately removes facade elements such as windows decorative features and fine edges and reconstructs the outline into simplified massing boxes. The higher HD quantitatively encodes the semantic transition from structural sketches to volumetric representations.

4.4 Conclusion

In this chapter, an automatic LoD sketch extraction framework based on generative artificial intelligence is proposed to automatically generate geometrically consistent and hierarchically coherent multi-LoD sketches from high-detail architectural models. The method integrates deterministic edge extraction with generative modeling, forming a progressive process from full-detail representation to volumetric abstraction. Through three key stages: (1) full-detail sketch extraction, (2) generative detail reduction, and (3)

volumetric abstraction, the framework achieves continuous simplification from LoD3 to LoD1 while maintaining structural proportions and spatial alignment. Experiments demonstrate that the proposed pipeline effectively removes redundant details, preserves geometric integrity, and produces visually coherent LoD sequences across various architectural types. Quantitative metrics (SSIM, MSE, and HD) confirm the balance between fidelity and abstraction at each level.

The multi-LoD sketch dataset constructed in this study provides a standardized foundation for AI-driven architectural generation, supporting multi-level model training and style transfer based on sketch inputs. As future work, the joint modeling of LoD sketches and BIM semantic data will be explored to enable an automated pipeline from sketch to BIM representation and to contribute to multi-level intelligent design and digital twin applications in architecture and urban environments. Building on the multi-LoD sketch representations established in this chapter, the next chapter focuses on multi-view architectural generation, where geometric consistency across viewpoints becomes a central modeling challenge.

Chapter 5

Multi-View Consistent Architectural Design

This chapter advances the building-level generative modeling workflow by addressing the problem of producing architectural representations that remain consistent across multiple viewpoints [24]. Using the LoD-aligned sketch and volumetric representations constructed in Chapter 4 as conditioning inputs, this study investigates how diffusion-based generative models can be guided by structural and depth-related cues to achieve coherent architectural imagery from simplified massing inputs.

In early-stage design practice, shoebox models serve as coarse abstractions of building massing, yet transforming these volumetric forms into detailed architectural images usually requires significant manual work. Generative artificial intelligence provides a promising direction for automating this transition, although maintaining stylistic and geometric consistency across multiple views remains a central challenge. To address this issue, this chapter introduces a multi-stage framework for generating architectural images from shoebox inputs, ensuring that the resulting representations remain aligned across different camera viewpoints.

The framework extends diffusion-based image generation by incorporating mechanisms that enforce cross-view coherence. ControlNet is adapted to process multi-view shoebox representations captured from fixed camera settings. An image-space consistency module is introduced to regulate style and structural alignment across views through the combination of style-based similarity, structural correspondence, and view-alignment constraints. To further enhance three-dimensional coherence, depth information is estimated from the generated images, and the paired image–depth data are used to refine spatial reasoning through a depth-aware attention mechanism.

The experimental results demonstrate that the proposed framework produces architectural images that exhibit strong cross-view consistency in both stylistic expression and geometric structure, even when generated from highly simplified volumetric inputs. These findings confirm that structural reasoning, initially developed at the object level, can be extended to architectural

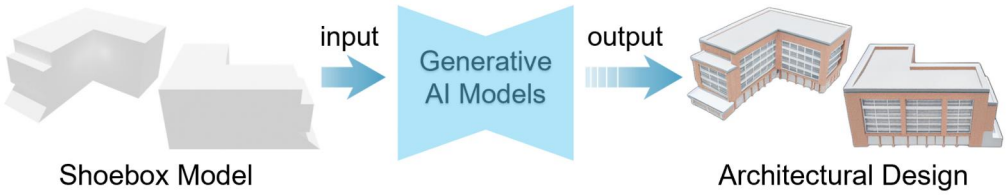


Figure 5.1: Research purpose and pipeline input–output overview.

forms through multi-view supervision and depth-aware modeling.

5.1 Background

Rapid progress in generative artificial intelligence has expanded the potential of architectural design [47]. Deep learning-based generative models can produce photorealistic architectural images with detailed rendering quality and reduce the labor-intensive process of translating early design intentions into visual form. These models support efficient evaluation of design concepts and have become important in contemporary architectural workflows [110].

In the early conceptual phases of design, shoebox models provide a practical means of expressing volumetric intent. Compared with sketches or textual descriptions, shoebox models deliver clearer indications of spatial layout and massing. However, converting these simplified representations into detailed architectural designs remains a demanding task. As illustrated in Figure 5.1, the aim is to streamline this process through generative models capable of producing architectural imagery directly from shoebox inputs.

A primary difficulty lies in achieving multi-view consistency. The goal is to generate images of the same building from different viewpoints while preserving geometric coherence and stylistic uniformity. Consistent depiction of spatial organization and facade structure is essential for multi-story buildings with complex configurations. Existing generative models often encounter inconsistencies across views that reduce the reliability of the produced designs.

To address these issues, a three-stage multi-view depth-consistent image generation framework is introduced. The method integrates generative modeling with multi-view consistency constraints and focuses on university buildings which present challenges such as floor alignment and repetitive facade elements. The framework produces multi-view architectural images that preserve style, structure and depth relationships from simplified shoebox inputs.

Research on multi-view consistency in image generation and 3D reconstruction provides analytical foundations for this work. Zero123++ [111] employs conditional diffusion models to produce viewpoint-consistent images from a single view using learned geometric priors. MVControl [112] improves viewpoint control by incorporating a hybrid diffusion prior to maintain structural and geometric stability across multiple views. DreamComposer [113] generates consistent images by leveraging multi-view features extracted from 3D representations and supports tasks that require controlled viewpoint and stylistic coherence. Although these studies demonstrate the feasibility of enforcing multi-view consistency, architectural image generation introduces additional constraints involving facade structure, window alignment and planar ordering.

Multi-modal fusion studies also inform the proposed framework. Combining RGB images with depth information enhances multi-view stability and strengthens structural consistency when generating complex scenes. MiDaS [114] estimates monocular depth that contributes to structural coherence in multi-view tasks. MVD-Fusion [115] fuses RGB and depth data to regulate geometric consistency in reconstruction pipelines. One-2-3-45 [116] integrates depth and viewpoint conditioning to reconstruct multi-view-consistent 3D models from single images. SyncDreamer [117] applies a 3D-aware attention mechanism to synchronize intermediate features across views and maintain structural similarity. These studies highlight the importance of combining depth and RGB information to support multi-view consistency.

The main contributions of this work are summarized as follows:

- A three-stage multi-view image generation framework that enables the generation of architectural designs from simple shoebox model inputs.
- An image-space loss module that complements the latent-space loss to improve the structural accuracy and stylistic consistency of multi-view images.
- A publicly accessible dataset of university teaching buildings that includes shoebox model images paired with corresponding architectural designs.

5.2 Methodology

A three-stage framework is proposed that progressively improves the structural and style consistency of the generated multi-view architectural images, as shown in Figure 5.2. Firstly, an improved multi-view diffusion model is proposed to generate intermediate architectural images from shoebox model inputs. Depth information is then estimated from the intermediate images

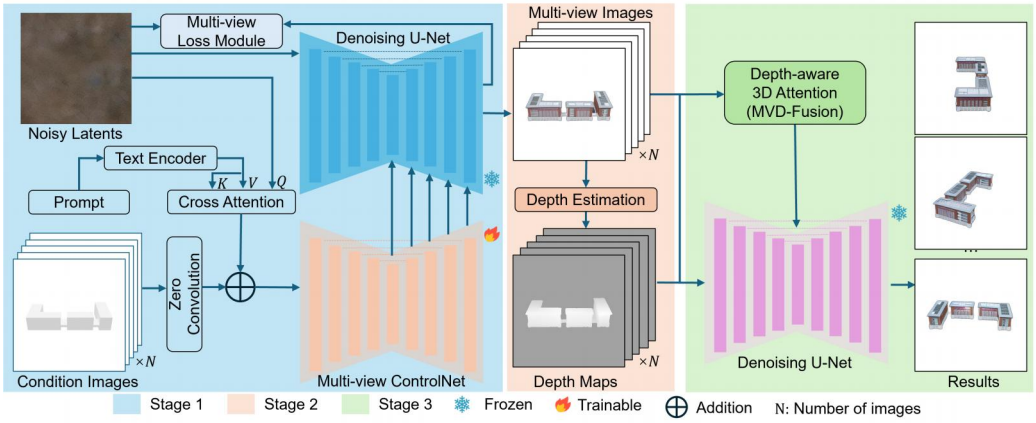


Figure 5.2: Overview of the proposed three-stage framework.

using monocular depth estimation method. Finally, the depth information is integrated with the intermediate architectural images through a multi-view fusion process.

5.2.1 Multi-View Diffusion Model

The proposed generation framework takes shoebox model images from the early architectural design stage as input and aims to generate multi-view architectural designs with color, texture, and structural details as output. The ControlNet [8] model is improved by introducing a multi-view module to achieve multi-view diffusion generation. Unlike single view to multiple image generation, this module can process multiple viewpoints simultaneously and ensure consistency across views.

As shown in the first stage in Figure 5.2, the multi-view ControlNet model is improved based on the stable diffusion model [7], which optimizes the single-view diffusion process into a multi-view generation module. To improve multi-view consistency, a cross-attention mechanism is incorporated. By embedding text features into noisy latent features and sharing features from different views, the generated design can more accurately follow the textual prompts (such as style description, architectural requirements, etc.), improving the generation quality and maintaining structural consistency and visual continuity across different views.

To optimize multi-view consistency in the generated images, in addition to the basic loss in the ControlNet latent space, a loss module in the image space is designed and incorporated, as shown in Figure 5.3. This module further improves multi-view consistency through three additional cross-view

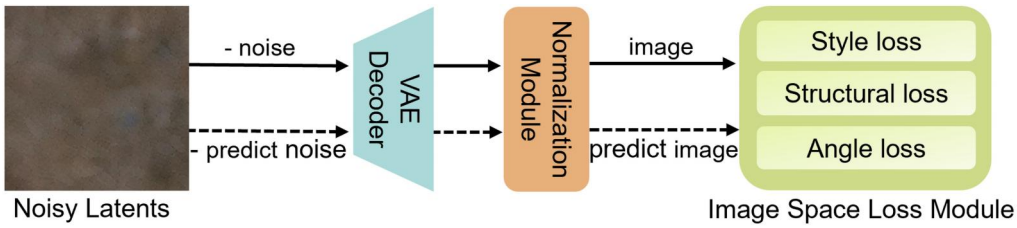


Figure 5.3: Multi-view consistency losses in image space.

constraint losses: style consistency loss, structural consistency loss, and angle alignment consistency loss.

The consistency of the style between the generated and the reference image is evaluated by calculating the difference in the Gram matrices of each viewpoint based on the Visual Geometry Group (VGG) [118] feature map. This loss ensures that the images generated from all viewpoints exhibit the same style.

$$L_{\text{style}} = \sum_{i=1}^{N} \|G(\phi(g_i)) - G(\phi(r_i))\|_2^2 \tag{5.1}$$

Where N is the number of viewpoints. g_i and r_i are the generated and real images in the same viewpoint. $G(\cdot)$ is the Gram matrix for capturing style features. $\phi(g_i)$ and $\phi(r_i)$ are the feature maps extracted by the VGG network.

The consistency of the geometric structure is guaranteed by two distinct types of loss: perceptual loss and multi-view content consistency loss. Perceptual loss compares the VGG features of the generated image and the reference image.

$$L_{\text{percep}} = \sum_{i=1}^{N} \|\phi(g_i) - \phi(r_i)\|_2^2 \tag{5.2}$$

The multi-view content consistency loss compares the feature differences between the generated image and the reference image in neighbouring viewpoints to ensure the content consistency of the generated image in multiple viewpoints.

$$L_{\text{content\_cos}} = \sum_{i=1}^{N-1} \left(\|\phi(g_i) - \phi(g_{i+1})\|_2^2 - \|\phi(r_i) - \phi(r_{i+1})\|_2^2\right)^2 \tag{5.3}$$

Angle alignment consistency loss compares the pixel differences between the generated image and the real image in adjacent viewpoints to ensure the consistency of the generated image and the real image in terms of geometric information across multiple viewpoints.

$$L_{\text{angle\_cos}} = \sum_{i=1}^{N-1} \left(\|g_i - g_{i+1}\|_2^2 - \|r_i - r_{i+1}\|_2^2 \right)^2 \tag{5.4}$$

Where $\alpha, \beta, \gamma, \delta$ denote the loss weights. The total loss optimizes style, structure, and angle consistency across views during training, resulting in architectural designs that maintain content and geometric coherence across multiple views.

$$L_{\text{total}} = \alpha L_{\text{style}} + \beta L_{\text{percep}} + \gamma L_{\text{content\_cos}} + \delta L_{\text{angle\_cos}} \tag{5.5}$$

5.2.2 Depth Estimation

Depth information estimation in real-world datasets often exhibit inconsistencies due to differences in data sources and acquisition methods. In addition, the inherent diversity of architectural designs, including variations in geometric structure and style, further complicates the challenge of maintaining consistency in depth information. To address the challenges posed by inconsistencies and biases in available depth data for architectural design, the framework adopts a depth estimation strategy based on multi-objective learning. This approach enables the integration of depth information from multiple datasets while mitigating the depth inconsistency problem. By improving structural consistency across different views, the depth estimation strategy supports multi-view coherence for diverse architectural designs.

In the implementation, MiDaS [114], a state-of-the-art monocular depth estimation tool, is utilized as a pre-trained model for extracting depth information. Depth features are extracted from the architectural designs generated in the first stage by employing a encoder with multi-scale feature extraction capabilities. These features are captured from each view of the design to generate high quality depth maps as complementary information to improve the structural consistency of the generated results.

5.2.3 Multi-View Fusion

In the third stage, the generated images and corresponding depth maps of each view are fed into MVD-Fusion [115], a multi-view depth consistency fusion framework, to verify and optimize the design with consistent style

and structure. By employing 3D reconstruction method based on multi-view depth consistency, this fusion mechanism effectively reduces depth differences between different views, and minimizes deformation issues during generation, and achieves robust feature fusion across multiple fields of view. Combining this multi-view fusion mechanism with feature alignment ensures the structural and stylistic consistency of the reconstructed 3D model and enables coherent multi-view presentation of the building.

Note that the main feature of the proposed method is adopting additional depth-aware cross-attention layers in the U-Net framework of ControlNet [8]. These attention layers are distributed throughout the various layers of the U-Net model, capturing cross-view depth features during encoding and decoding to optimize perceptual feature consistency among multi-view images. The residual cross-attention layer dynamically maps features from different views into a shared depth latent space. The cross-attention mechanism can process multi-view input in parallel, learning the feature relationships between different views to represent the geometric information of the building structure consistently across views.

5.3 Experiments

5.3.1 Training Dataset

A specialised dataset was constructed to generate detailed architectural designs from architectural shoebox models. This work began by conducting an extensive survey of university teaching buildings in China, analyzing their architectural characteristics and features. Based on this survey, models of university buildings were collected and their essential structural elements were captured to create simplified shoebox models for these buildings. This effort resulted in a comprehensive paired 3D dataset containing 210 shoebox models and their corresponding detailed models.

Given the large amount of data that needs to be processed when working with 3D models, the multi-view design generation task was decoupled into a simple image-to-image generation task. Based on the initial dataset, Blender was used to render from multiple angles and generate a multi-view image dataset for training and testing. To include the roof features, a 30-degree perspective angle was used, with a 6-degree rotation applied to generate 60 views per model. A total of 12,600 images of the shoebox models and 12,600 images of the building model details were obtained.

5.3.2 Implementation Details

The model was trained on a high-performance computing server equipped with an NVIDIA A100 GPU (80GB) and CUDA 12.5, to meet the computational demands of large-scale multi-view architectural generation tasks. The loss weights $\alpha, \beta, \gamma,$ and δ were set to $10^9, 100, 1,$ and 10, respectively, to balance the importance of each loss term. The training took approximately 200 hours and around 500,000 iterations.

5.3.3 Reconstruction and Generation Experiments

After completing the model training, two types of experiments were conducted to comprehensively assess the performance of the proposed framework: reconstruction experiments and generation experiments. The reconstruction experiments were designed to evaluate how well the model adapts to data seen during training. A subset of architectural shoebox images from the training dataset was provided as input, and the model was used to generate the corresponding architectural design images. The reconstruction results are presented in Figure 5.4. In contrast, the generation experiments examined the model's ability to generalize to previously unseen data. Shoebox model images that were not included in the training set were used as input, and the model produced architectural design images based on these novel samples. The results of the generation experiments are shown in Figure 5.5.

5.3.4 User Evaluation

To evaluate the performance and practical applicability of the model in generating consistent and high-quality architectural designs, a user evaluation was conducted with 15 graduate students (9 male and 6 female) from the Department of Architecture. Each participant was presented with 14 reconstructed models and the corresponding 14 target models, as well as 10 generated models, each with 6 images representing different perspectives. The study assessed the effectiveness of the model-generated architectural designs in real-world applications across six core dimensions: structural integrity, structural consistency, detail integrity, detail consistency, visual aesthetics, and practicality. Participants used a 5-point scale (1 for strongly disagree, 5 for strongly agree) to rate the generated images on these dimensions. The analysis of the user evaluation results is shown in Figure 5.6.

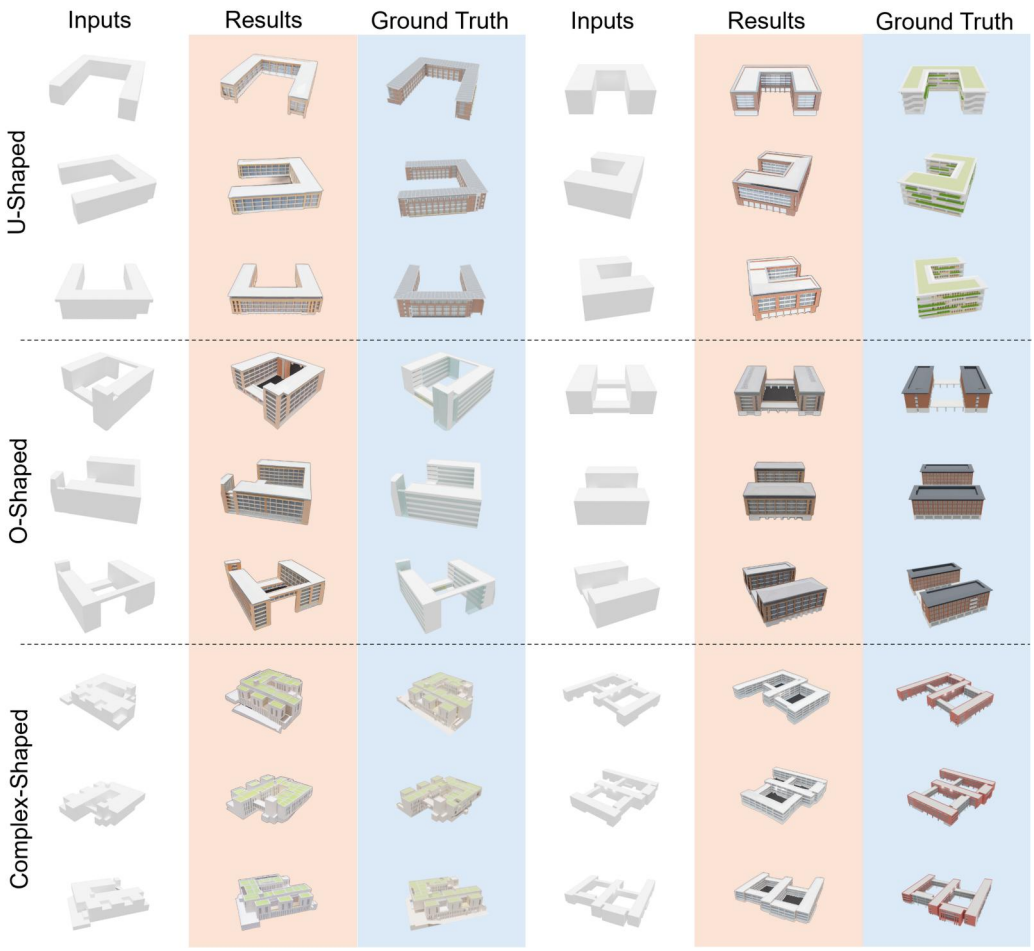


Figure 5.4: Reconstruction results for U-, O-, and complex-shaped university buildings.

5.4 Results

5.4.1 Reconstruction Results

Figure 5.4 presents the reconstruction results. The proposed model is able to accurately reproduce the detailed features of the architectural designs, while maintaining strong geometric and structural consistency across the reconstructed views. Moreover, the model effectively captures textural information, resulting in visually coherent and realistic outputs. These findings demonstrate the capability of the model to handle complex architectural configurations, including multi-storey facades and compositions involving multiple shoebox volumes, without compromising consistency across view-

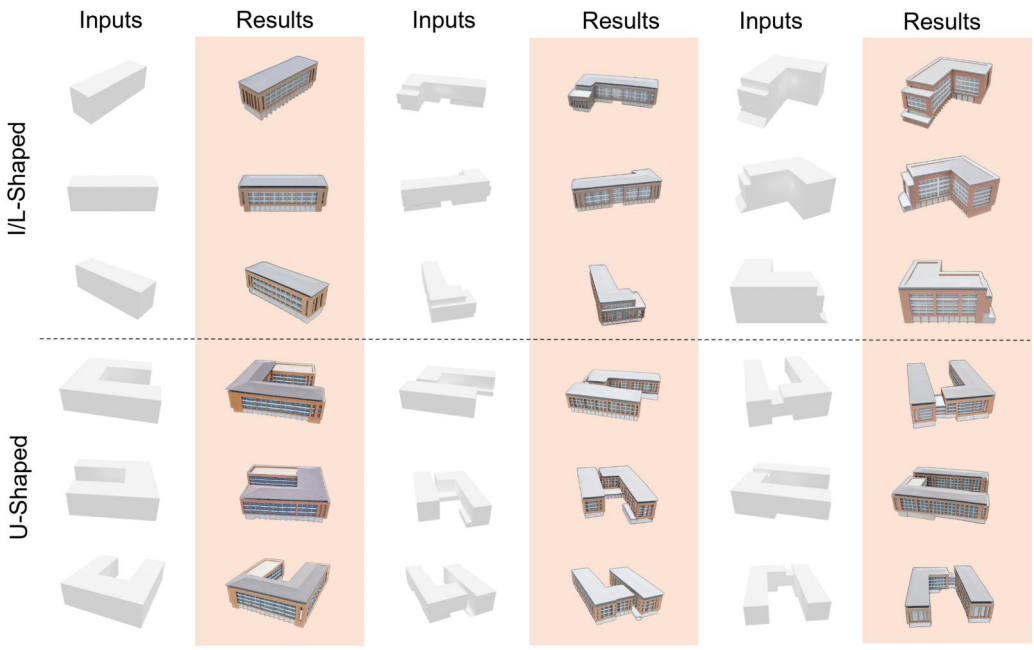


Figure 5.5: Generation results for I/L-shaped and U-shaped university buildings.

points. Overall, the reconstruction results indicate that the model adapts well to the training data and can generate high-quality outputs suitable for architectural visualization tasks.

5.4.2 Generation Results

Figure 5.5 shows the results of the generation experiment, which shows that the model can generate realistic and consistent images of architectural designs, and has strong generalisation capabilities for unknown data. The multi-view images generated are consistent in style and structure, and excel in the geometric alignment of architectural structures and the rendering of texture details.

5.4.3 User Evaluation Results

Figure 5.6 presents the statistical analysis of the evaluation results. For the reconstruction task, the average scores for structural integrity ($M = 3.567$, $SD = 1.234$) and structural consistency ($M = 3.533$, $SD = 1.324$) suggest a strong level of multi-view stability and geometric accuracy. Similarly, the scores for detail integrity ($M = 3.557$, $SD = 1.099$) and detail

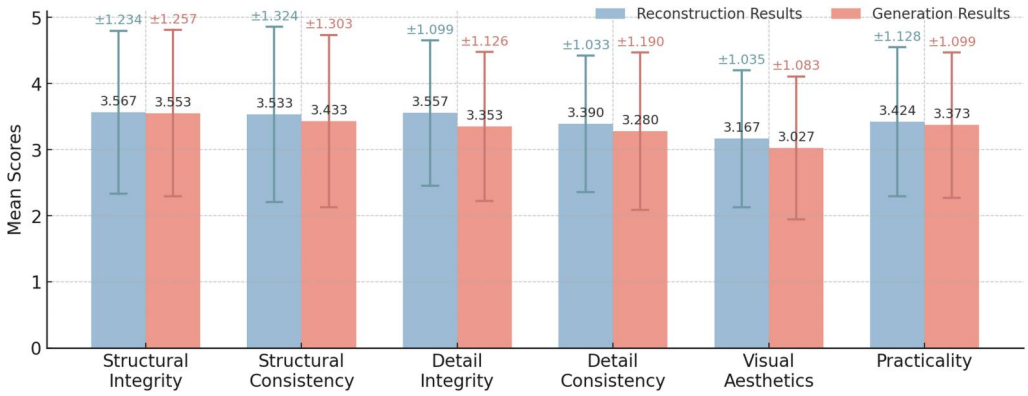


Figure 5.6: User evaluation results (means and standard deviations per metric).

consistency ($M = 3.390$, $SD = 1.033$) indicate reliable preservation of fine-grained features across different viewpoints. Although the scores for the generation task are slightly lower, they still demonstrate encouraging performance. Structural integrity ($M = 3.553$, $SD = 1.257$) and structural consistency ($M = 3.433$, $SD = 1.303$) remain comparable to the reconstruction results, showing that the model maintains robustness even when handling previously unseen inputs. For detail integrity ($M = 3.353$, $SD = 1.126$) and detail consistency ($M = 3.280$, $SD = 1.190$), the model demonstrates reasonable accuracy in capturing local architectural details, though the results also indicate potential for improving fine-detail preservation. With respect to visual aesthetics and practical usability, both reconstruction and generation tasks received slightly lower ratings than the structural and detail-related metrics, yet still reflect moderate success. Reconstruction results achieved mean scores of $M = 3.167\,(SD = 1.035)$ for visual aesthetics and $M = 3.424\,(SD = 1.128)$ for practicality, indicating a solid balance between visual quality and functional plausibility. Generation results showed mean scores of $M = 3.027\,(SD = 1.083)$ for visual aesthetics and $M = 3.373\,(SD = 1.099)$ for usability, highlighting the model's potential for further refinement while demonstrating its capability to produce usable outputs for novel input cases.

5.5 Conclusion

This chapter presents a novel multi-view–consistent image generation approach based on generative AI models, with a focus on producing coherent

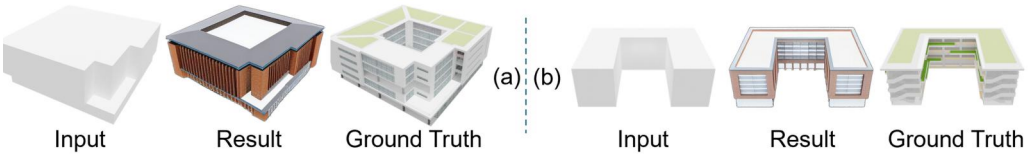


Figure 5.7: Examples of Structural Mismatch in Reconstruction.

multi-view architectural renderings from simplified shoebox representations. Experimental results demonstrate that the proposed method offers clear advantages in maintaining cross-view consistency, preserving detailed geometric features, and delivering high-quality visual outputs. Despite these strengths, the method also exhibits several limitations. First, the shoebox models used in the experiments are relatively simple and do not fully capture the diversity and complexity of real-world architectural inputs. This simplicity may restrict the model's generalization capability when processing more intricate building geometries. For example, as shown in Figure 5.7(a), the extracted shoebox model fails to represent the hollow central structure of the building, and in Figure 5.7(b), the shoebox abstraction does not adequately capture the stair-related geometry. Such discrepancies can lead to mismatches between the generated images and the intended architectural structures. Future work will therefore focus on broadening the range of input datasets beyond university buildings, incorporating diverse architectural typologies, facade styles, and material variations. Additionally, further improvements will target more precise control over fine-grained architectural details. Through these extensions, the goal is to move the framework beyond a proof-of-concept stage and toward practical solutions suitable for adoption in real-world architectural design and production workflows. Together with the LoD-aligned representations established in Chapter 4, this chapter forms the generation component of the building-level pipeline, where controllable structure and view-consistent appearance are both required for practical design use. The next chapter extends the generative framework from architectural synthesis to architectural renovation. Chapter 6 introduces a sketch-based and text-driven facade renovation pipeline that enables the modification of existing buildings while preserving their structural semantics and overall design logic.

Chapter 6

Architectural Facade Renovation

This chapter focuses on facade renovation as a constrained form of building-level generative modeling, where new design elements must be introduced while preserving the existing architectural structure [25]. Building on the structural alignment mechanisms introduced in Chapter 4 and the multi-view generative capabilities developed in Chapter 5, it examines how generative artificial intelligence can integrate sketch inputs, textual descriptions, and geometric context to produce renovation proposals that remain consistent with the existing architectural structure and design intent.

Facade renovation provides a more sustainable alternative to full demolition, yet producing design proposals that preserve the original structure while expressing new intent remains a demanding task. Conventional workflows require detailed as-built modeling before design exploration, which is both time-consuming and labour-intensive. To address these limitations, this chapter proposes a three-stage framework that combines generative artificial intelligence with vision-language models to support controlled and interpretable facade renewal. The framework begins with the use of a fine-tuned vision-language model to analyse a rough structural sketch and textual descriptions, predicting the locations of modifiable regions and the types of elements to be introduced. It then employs a diffusion-based generator to produce detailed sketches of the new components, which are combined with the existing outline through a generative inpainting procedure. In the final stage, ControlNet is applied to refine the integrated sketch into a coherent and photorealistic facade image.

Experimental evaluations conducted on curated datasets and real industrial buildings show that the proposed framework can generate renovation proposals that preserve the original volumetric structure while enhancing facade detail quality. The method effectively bypasses the need for detailed as-built modeling and enables architects to explore alternatives rapidly, iterate during the early stages of design, and communicate renovation intentions more efficiently.

By integrating structural cues, textual semantics, and multi-modal generation processes, this chapter complements the earlier studies on architectural

image generation and completes the building-level layer of the dissertation. It also provides the conceptual link to the city-level generative forecasting developed in the following chapter.

6.1 Background

Industrial heritage revitalization has long emphasized the importance of preserving structural identity while enabling contemporary adaptation. Foster argued that industrial buildings can be given renewed life through careful design interventions that respect both existing fabric and new functionality [119]. Compared with demolition and reconstruction, adaptive reuse and facade renovation provide more sustainable and economically viable pathways for updating industrial facilities. Traditional workflows, however, follow linear and labor-intensive processes that move from conceptual sketches to 3D models and construction documentation. These stages demand repeated revisions, intensive manual work and substantial communication among stakeholders [62].

Recent developments in generative AI have introduced new possibilities for architectural renovation. Generative models are becoming promising tools for accelerating early design exploration and for producing realistic visualizations during the conceptual phase [47]. Despite this potential, applying generative models directly to facade renovation presents distinct challenges. Many existing models struggle to interpret architectural semantics and cannot automatically determine which portions of the facade should be modified, preserved or replaced. This limitation makes it difficult to translate design intent into spatial actions. Although diffusion-based image generation can produce new facade elements, structural coherence and stylistic continuity between modified regions and the original context remain difficult to maintain [7, 29]. Generated results often lack precise details and photorealistic textures, reducing their suitability for professional communication.

To address these issues, a three-stage generative framework is developed that integrates vision–language models for semantic understanding of sketches and textual descriptions and combines these with Stable Diffusion, IP-Adapter and ControlNet to produce contextually coherent renovation proposals. The overall architecture of this framework is illustrated in Figure 6.1. The study focuses on the adaptive transformation of industrial factories into commercial buildings and aims to retain industrial character while enhancing facade composition and visual quality. Experiments and case studies using real industrial building data demonstrate that the method preserves

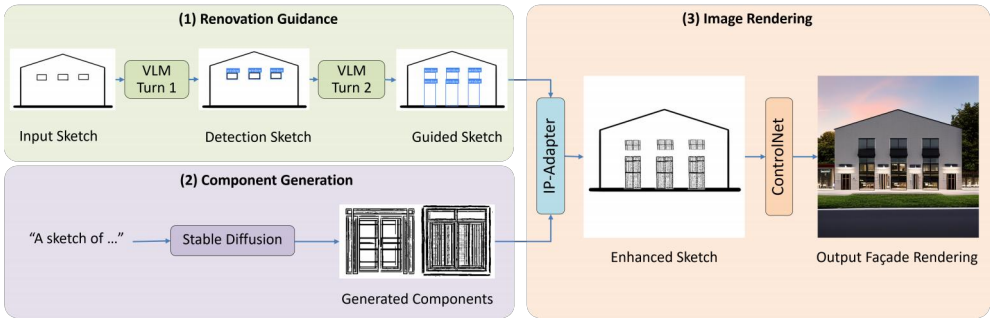


Figure 6.1: Overview of the proposed three-stage facade renovation framework.

structural integrity, enriches facade articulation and achieves photorealistic realism, offering a practical solution for industrial building renewal.

Research in architectural image generation and facade renovation provides essential context for this framework. Text-to-image diffusion models including Stable Diffusion [7] support the synthesis of high-quality images from text descriptions. Conditional architectures such as ControlNet [8] introduce structural priors including edges and depth maps to refine the generative process. Image translation methods including Pix2Pix [29] have been applied to facade editing and architectural style transformation. Adapter-based approaches such as IP-Adapter [76] extend controllability by integrating external visual references. Despite these developments, most generative models remain disconnected from actual architectural renovation practice and tend to emphasize aesthetic synthesis rather than structural constraints or early-stage design processes. Studies of adaptive reuse in heritage building contexts focus on preserving historical character and improving functional and environmental performance [120]. However, these workflows remain dependent on manual modeling, professional expertise and deterministic evaluation.

This gap indicates the need for data-driven generative methods that can automate early design exploration and integrate sketch-conditioned inputs with structured generative pipelines. The proposed framework addresses this need by combining vision-language interpretation with diffusion-based synthesis in order to produce facade renewal proposals that preserve geometric and semantic integrity while offering new design possibilities.

Vision–language models (VLMs) provide an additional foundation for this approach. A vision–language model is a multi-modal framework that jointly processes visual and textual information through paired encoders. Early work including CLIP [9] demonstrated shared latent embedding spaces

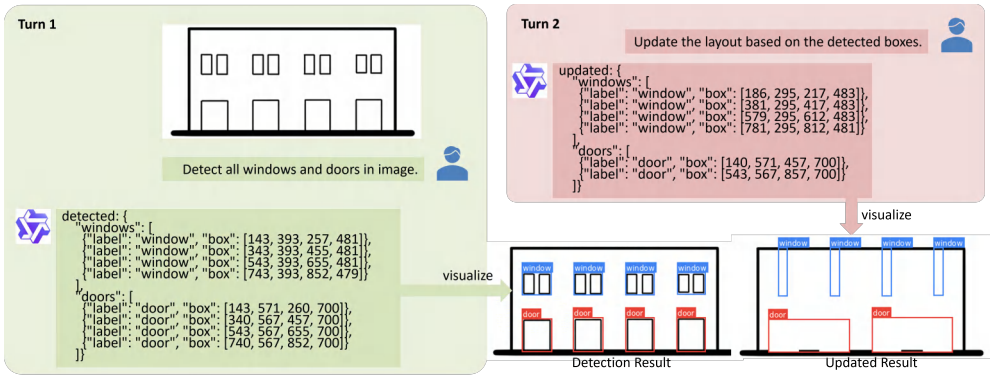


Figure 6.2: Pipeline of the two-turn supervised fine-tuning process for the VLM.

for image and text that support cross-modal prediction. Recent systems including Qwen3-VL [121] combine large-scale vision encoders with advanced language processing capabilities to support spatial grounding and semantic interpretation. In this study, a fine-tuned Qwen3-VL model interprets rough architectural sketches and textual descriptions and guides the generation process toward renovation results that align with both structural logic and semantic intent.

The main contributions of this study are summarized as follows:

- This work proposes a three-stage generative framework that integrates sketches and textual descriptions to automate facade renovation design.
- This work constructs a publicly accessible dataset of industrial building sketches, consisting of paired original sketches and renovated sketch outputs.
- This work demonstrates that incorporating a VLM with architectural sketches enables both textual and visual reasoning, producing meaningful design guidance for facade renovation.

6.2 Methodology

6.2.1 Renovation Guidance via Vision–Language Model

The proposed framework takes as input a rough structural sketch that preserves the overall geometry and spatial proportions of a building. The goal is to produce textual renovation guidance that describes where modifications should occur and what architectural components such as windows entrances or facade features should be added. To achieve this, Qwen3-VL-

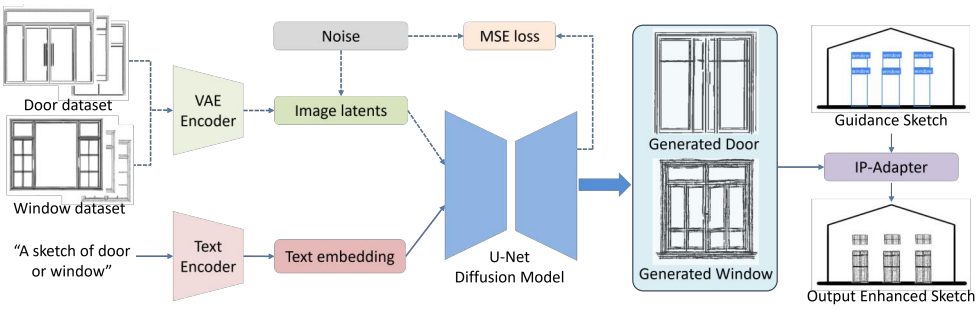


Figure 6.3: Diffusion-based component generation and sketch enhancement pipeline.

4B-Instruct [121], a large-scale multimodal vision–language model composed of a visual encoder, a multimodal projection module, and a language decoder, is fine-tuned.

To adapt the model to architectural sketch understanding, a parameter-efficient supervised fine-tuning strategy is adopted. The visual encoder and the language model remain frozen, while only the multimodal projection layer is updated. This approach allows the model to learn domain-specific visual–textual alignment while mitigating overfitting and reducing computational overhead.

Each training instance is constructed as a two-turn instruction–response conversation, simulating an interactive process between a designer and an intelligent assistant. The two-turn paradigm consists of:

- **Detection phase**: the model detects existing structural components based on the instruction (e.g., "Detect all windows and doors in the image"). It performs spatial analysis and produces a textual description of facade elements and spatial zones.
- **Guidance phase**: conditioned on the previous analysis, the model receives a follow-up instruction (e.g., "Update the layout based on the detected boxes") and generates targeted renovation suggestions.

Through this two-step reasoning process, the VLM learns to associate structural interpretation with renovation actions, enabling coherent and spatially consistent textual guidance for facade modification. An overview of this fine-tuning pipeline is shown in Figure 6.2.

6.2.2 Component Generation and Sketch Enhancement

Building upon the renovation guidance generated in the first stage, the second stage focuses on synthesizing new architectural components and enhancing

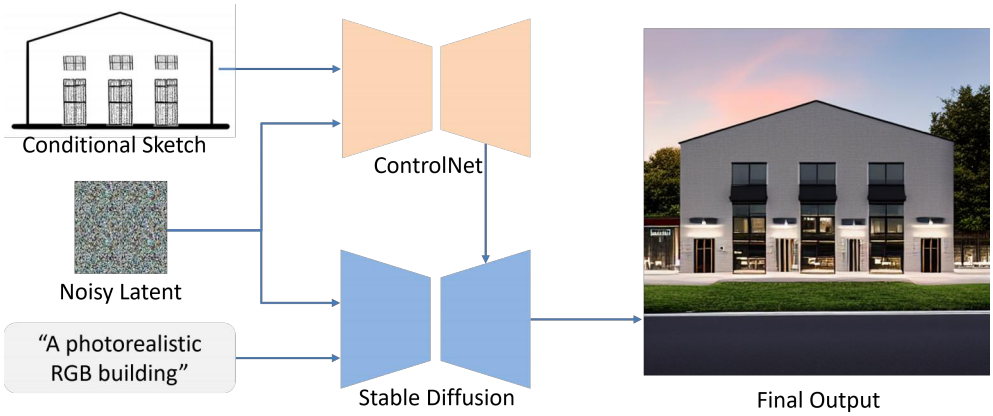


Figure 6.4: Photorealistic image generation guided by sketch-based structural conditioning.

the original sketch through a diffusion-based generation pipeline. As shown in Figure 6.3, a fine-tuned Stable Diffusion model [7] serves as the generative backbone to synthesize components such as windows and doors.

Guided by the textual renovation descriptions obtained from the VLM, the diffusion model generates components that are structurally aligned and stylistically consistent with the original facade. This ensures that newly introduced elements correspond precisely to the modification regions identified in the guidance stage.

To integrate these generated components into the building outline, a pretrained IP-Adapter inpainting pipeline [76] is employed. The IP-Adapter augments the diffusion model with image-prompt conditioning, enabling localized, seamless blending between new and existing structures. The resulting enhanced sketch preserves geometric integrity while accurately reflecting the intended renovation outcome.

6.2.3 Photorealistic Architecture Image Generation

In the final stage the enhanced sketch, which combines the original structural outline with newly generated components, serves as the structural condition for photorealistic image synthesis. This stage aims to transform the conceptual renovation sketch into a high fidelity architectural visualization suitable for design assessment and presentation.

As illustrated in Figure 6.4, a pretrained Stable Diffusion model [7] is adopted as the generative backbone, combined with ControlNet [8] to enforce spatial conditioning during diffusion sampling.

ControlNet ensures that the generated images remain faithful to the

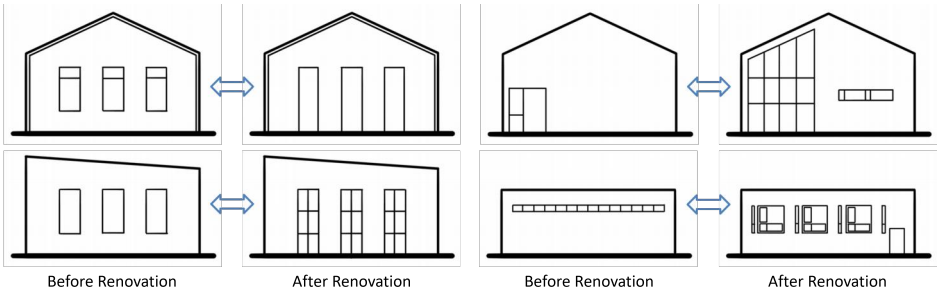


Figure 6.5: Examples from the facade renovation dataset used for VLM fine-tuning.

underlying sketch by constraining the sampling trajectory to closely follow the geometric structure of the input condition. This enables the synthesis of realistic facade images with appropriate materials, shadows, textures, and lighting effects, while preserving the architectural correctness of the design.

The resulting renderings bridge the gap between conceptual sketches and professional visualization, enabling high-quality outputs suitable for architectural review, communication, and decision-making.

6.3 Experiments

6.3.1 Training Dataset

Facade Renovation Dataset. To fine-tune the VLM for renovation guidance, a specialized facade dataset was constructed focusing on industrial buildings in Tianjin, China, primarily built between the 1950s and 1980s. Building references were collected from publicly available imagery and renovation case archives, providing representative examples of industrial facade transformations. Based on these references, 100 paired facade samples (Figure 6.5) were manually created, each consisting of a "before-renovation" and an "after-renovation" image that correspond in geometry and transformation intent.

Considering the variations in scale and roof morphology among heavy industrial factories, the dataset was categorized by building size and roof type. All selected cases represent steel-structure industrial buildings, which are generally in sound structural condition and offer high renovation potential. The buildings range from 10–20 meters in width and 5–10 meters in height, encompassing both pitched and flat roof configurations.

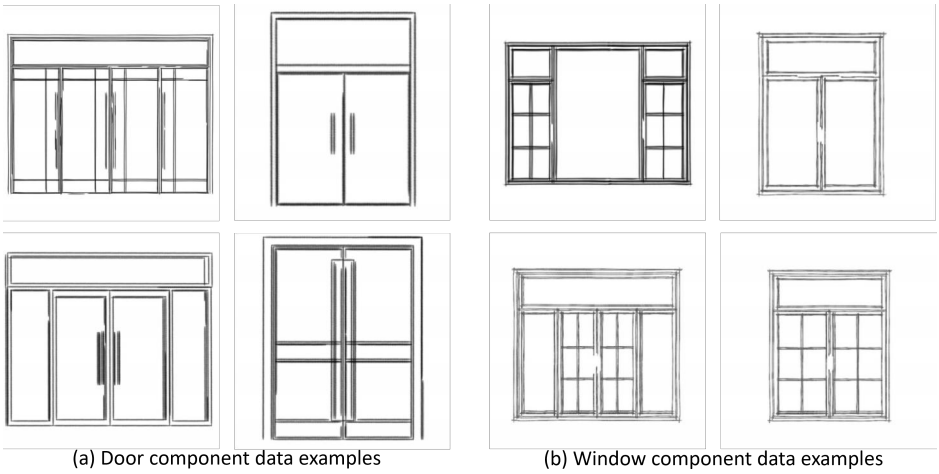


Figure 6.6: Examples from the component dataset used for diffusion-based generation.

Component Dataset for Sketch Enhancement. To support the generation of architectural components during sketch enhancement, an auxiliary component dataset was also developed. As shown in Figure 6.6, this dataset comprises 107 door sketches and 164 window sketches, which serve as targeted training samples for the diffusion-based component generation model described in Section 3.2. Each sketch was manually drawn to capture representative geometries and facade element styles commonly found in industrial renovation projects.

6.4 Experiments

To evaluate the overall performance of the proposed three-stage generative framework, two types of experiments were conducted: reconstruction and generation. All experiments follow the same three-stage pipeline described in Section 3. Representative visual results are shown in Figures 6.7 and 6.8.

Reconstruction Experiments. In the reconstruction experiments, a subset of sketches from the training dataset was used as input to evaluate the framework's ability to reproduce design outcomes consistent with the training distribution. The VLM first generated textual renovation guidance, describing where and how facade components should be modified. Subsequently, the diffusion model synthesized corresponding architectural components (e.g., doors and windows) based on the VLM's suggestions,

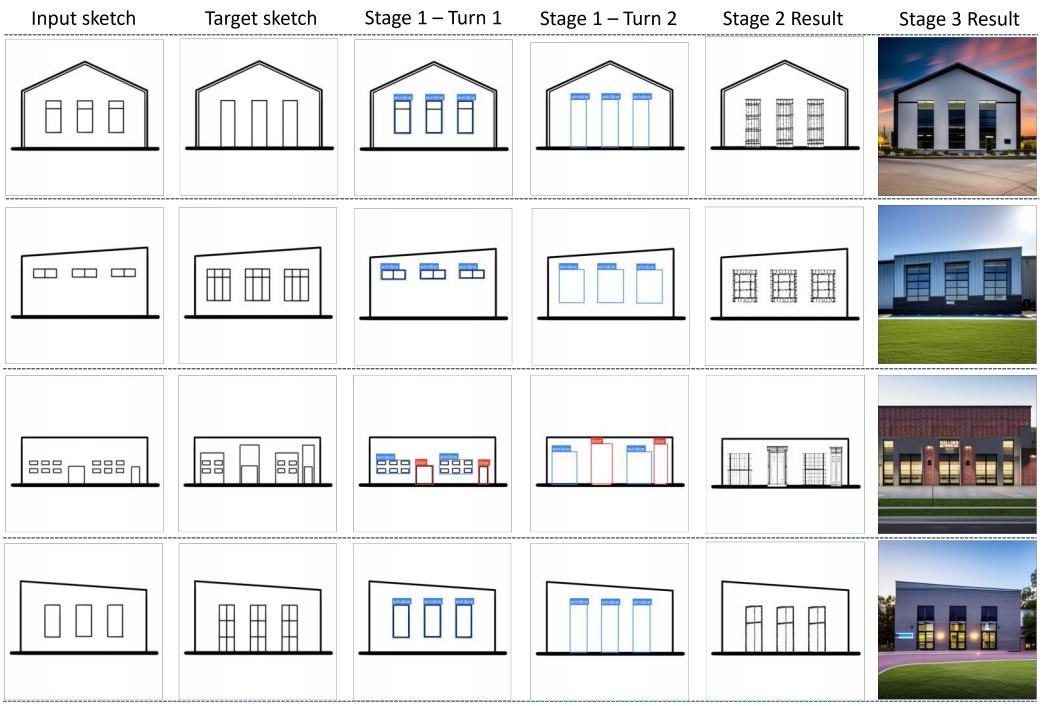


Figure 6.7: Reconstruction results of the three-stage framework.

enhancing the input sketches. Finally, the ControlNet refined the enhanced sketches into photorealistic facade renderings.

This end-to-end reconstruction process validates the framework's internal coherence, from semantic reasoning to visual realization, and demonstrates its ability to reproduce the geometric and stylistic consistency of the training data. Representative results are shown in Figure 6.7.

Generation Experiments. In the generation experiments, sketches that were not included in the training process were used as input to assess the framework's generalization ability. Following the same three-stage pipeline, the framework generated renovation suggestions, synthesized new facade components, and rendered photorealistic results.

This experiment demonstrates that the proposed framework can generalize beyond the training distribution, producing design outcomes that are both semantically consistent and visually coherent for unseen building layouts. The generation results are presented in Figure 6.8.

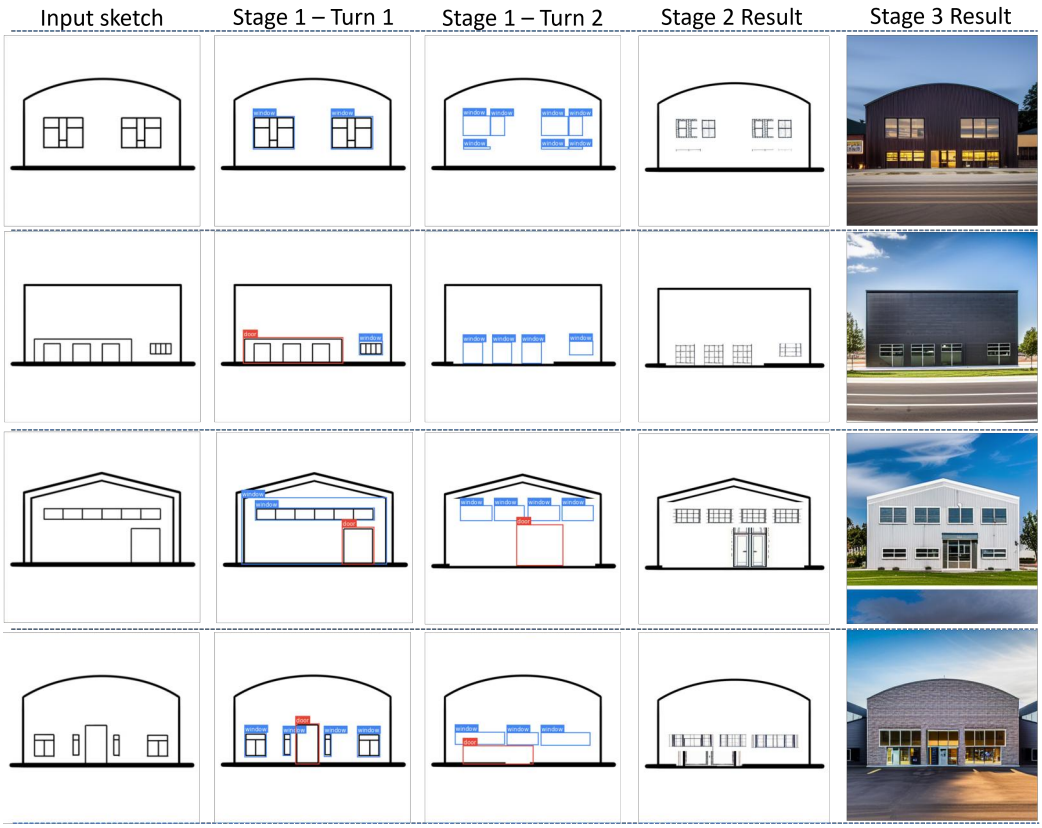


Figure 6.8: Generation results of the three-stage framework on unseen sketches.

6.5 Results

6.5.1 Reconstruction Results.

As shown in Figure 6.7, the reconstruction results demonstrate that the proposed framework accurately reproduces renovation layouts that align with the geometric configuration of the original sketches. The generated outputs exhibit structural fidelity, with newly added components conforming to the existing facade proportions and spatial hierarchy. The results confirm that the fine-tuned VLM provides accurate semantic guidance, while the diffusion model and ControlNet collaboratively ensure visually realistic reconstruction from sketches to rendered images. These findings validate the framework's capability to maintain both semantic coherence and visual consistency across all stages of generation.

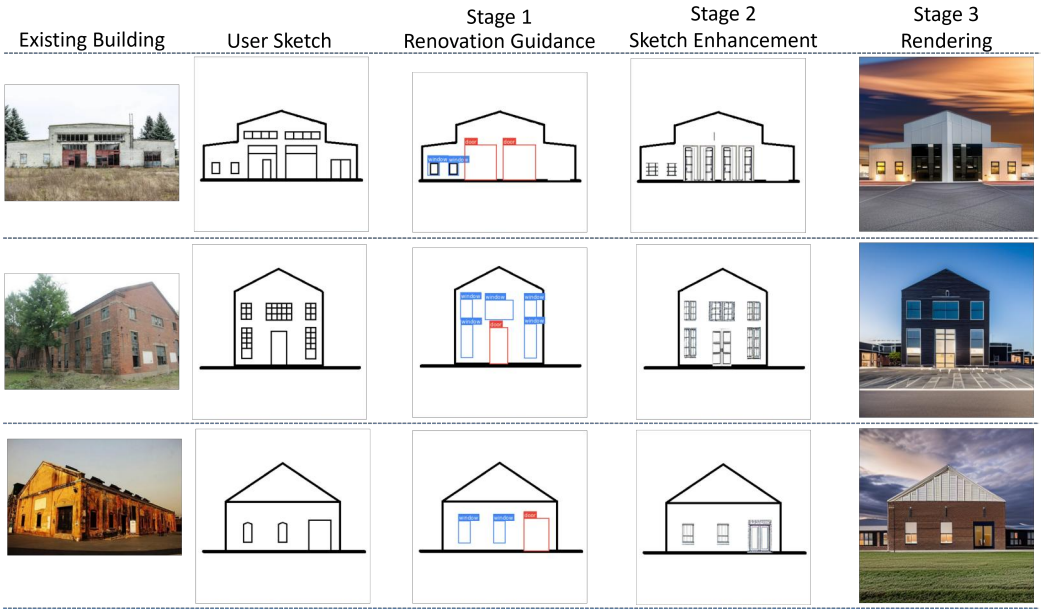


Figure 6.9: Real-world case study results produced by the proposed framework.

6.5.2 Generation Results.

As illustrated in Figure 6.8, the generation experiments demonstrate that the framework generalizes effectively to unseen architectural sketches. The model generates plausible and consistent renovation proposals, with component placement and scale that remain aligned with the structural logic of each sketch. This confirms that the combination of semantic reasoning (VLM), component synthesis (Stable Diffusion), and visual refinement (ControlNet) enables the framework to handle new building configurations while producing visually convincing renovation outcomes.

6.5.3 Real-World Case Studies.

To evaluate the practical applicability of the proposed three-stage generative framework, photographs of existing industrial buildings were collected and manually converted into structural sketches that capture their primary geometric outlines and facade features. These sketches were then processed through the proposed generation pipeline, which produced renovation suggestions based on the model's learned semantic and spatial understanding.

As shown in Figure 6.9, the framework demonstrates strong adaptability to real-world architectural conditions. The generated renovation guidance

aligns closely with the existing facade geometry and architectural composition, maintaining spatial coherence while producing design suggestions that are both visually plausible and structurally consistent. These results highlight the framework's potential as a practical tool for facade renovation and architectural visualization in real-world design contexts.

6.6 Conclusion

A novel sketch-based generative AI framework is proposed that transforms renovation concepts for existing industrial buildings into high-fidelity photorealistic visualizations, eliminating the need for time-consuming as-built modelling. Experimental results show that the proposed method has significant advantages in terms of preserving structural integrity, enhancing facade detail, and improving visual realism.

However, some limitations remain. The fine-tuned VLM was trained primarily on simplified, front-facing building sketches, which constrains its ability to handle more complex geometries, oblique perspectives, and varied architectural styles. Moreover, while the model can suggest plausible renovation concepts, it lacks engineering reasoning, meaning it does not consider constraints such as load-bearing columns or beams, which may lead to impractical design suggestions.

Future work will focus on expanding the dataset to include more diverse architectural typologies and viewpoints, as well as integrating structural and contextual understanding into the VLM-guided design process. Through these extensions, the goal is to advance the framework beyond the proof-of-concept stage toward a robust, deployable system that can support real-world architectural design and renovation workflows.

Together with the object-level and building-level generative mechanisms developed in the preceding chapters, the renovation framework presented here completes the building-level modeling stage of the dissertation. The next chapter extends this cross-level paradigm from individual buildings to city-level spatial prediction. Chapter 7 introduces a multi-conditional urban evolution model that integrates historical layouts, density patterns, height distributions, and contextual information to forecast long-term urban development. In contrast to the building-focused frameworks in Chapters 3–6, the urban model addresses macroscopic spatial dynamics, thereby completing the dissertation's progression from object-level geometry to building-level design and finally to city-level generative forecasting.

Chapter 7

Multi-Conditional Urban Evolution Forecasting

This chapter positions urban evolution forecasting as the largest scale component of the dissertation's cross-level generative modeling approach. Extending the structural and semantic modeling principles established at the object and building levels, this chapter examines how generative AI can represent and predict long term spatial change within complex and dynamic urban systems [26]. While the previous chapters addressed part level structure and building level reasoning across multiple levels of detail, the present study shifts the focus to city level processes and integrates multiple urban indicators including density, building height, transportation networks, and historical development patterns into a unified predictive framework.

This chapter represents the city level of the dissertation and addresses planning oriented challenges that emerge when generative models are applied to metropolitan development. The proposed MMCN framework combines multi conditional control, neighbor aware spatial memory, and temporal evolution modeling. Through this integration the study demonstrates how generative AI can capture regional coherence, spatial continuity, and long term developmental tendencies across large urban areas. The framework is designed not only to improve predictive accuracy but also to support practical planning scenarios that require reliable and interpretable models of urban growth.

Sustainable urban development requires predictive models that can integrate multiple interdependent factors such as building density, height distribution, transportation networks, and historical evolution in order to support evidence based planning. However, existing AI based generative models are limited in their ability to combine these factors, and their predictions often remain fragmented and insufficient for comprehensive urban planning strategies.

To address this limitation, MMCN (Memory aware Multi Conditional generation Network) is introduced as an AI driven framework that forecasts urban layout evolution by modeling the complex interactions among multiple

urban development variables. MMCN responds to three major planning challenges through technical innovations. First, a multi conditional control architecture processes density, height, and transportation conditions as interconnected components that influence sustainable urban form. Second, a spatial continuity mechanism ensures that generated layouts maintain regional coherence. Third, a temporal consistency component leverages historical layout patterns to capture long term evolution trends.

Using the comprehensive Shenzhen urban evolution dataset, the experiments show that MMCN improves forecasting accuracy and spatial coherence, achieving an SSIM of 0.885 and a Boundary IoU of 0.642 with clear advantages over all baseline models. This work provides technical support for planning practitioners engaged in sustainable development scenario analysis and enables the formulation of long term planning decisions that balance urban growth with sustainability objectives.

7.1 Background

With more than 68% of the global population expected to live in urban areas by 2050 [122], cities face significant challenges in balancing rapid growth with sustainability objectives. Urban areas consume roughly 75% of global energy and produce more than 70% of global greenhouse gas emissions [123]. Sustainable urban development is therefore central to achieving global climate goals and the Sustainable Development Goals of the United Nations, particularly SDG 11 on sustainable cities and communities [124]. The spatial structure of cities, including density, height distributions and transportation networks, plays a decisive role in shaping energy use, mobility patterns and overall urban quality of life [125]. Effective planning requires tools capable of predicting how current interventions will influence long-term spatial evolution.

As shown in Figure 7.1, urban development is driven by interdependent factors such as density regulations, height restrictions, transportation infrastructure and historical growth patterns [126]. These relationships are highly nonlinear and often difficult to anticipate through traditional planning approaches that rely on zoning practices and expert judgment [127]. Cities also hold extensive historical development records that could support data-driven forecasting, yet these datasets are rarely used systematically for long-term predictive analysis [128].

Recent advances in generative AI have shown strong capabilities in visual synthesis [7], yet several obstacles limit their use for urban planning. Current generative models typically treat density, height and transportation

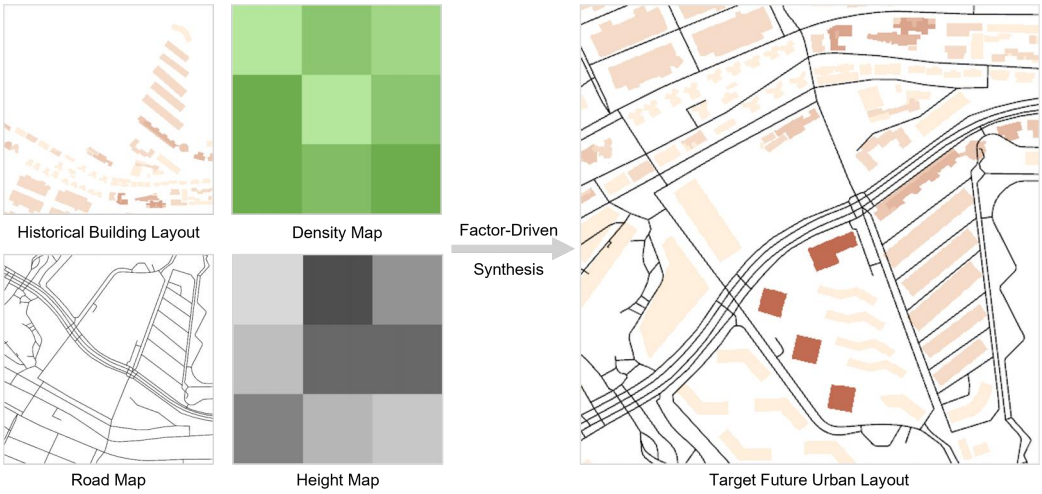


Figure 7.1: Overview of factor-driven urban layout modeling. The target urban layout is generated by incorporating multiple interdependent urban planning factors, such as building density, height map, road structure, and historical building patterns.

conditions as independent inputs rather than modeling their mutual dependencies [129]. Many systems also rely on patch-based generation without mechanisms that maintain spatial continuity across tile boundaries [130], leading to fragmented and inconsistent layouts. Moreover, most methods focus on single-time-step generation and lack the ability to capture temporal evolution patterns essential for long-term planning [131]. These issues restrict the practical usefulness of generative AI in sustainable planning scenarios.

At the methodological level, the evolution of generative models provides important context for this work. Early studies explored Variational Autoencoders for floorplans and functional distribution maps, as in BlockPlanner [73] and the Generative Isovist Transformer [74]. Generative Adversarial Networks expanded this line of research, enabling layout synthesis at architectural and district scales. Representative approaches include House-GAN [72], FloorplanGAN [60] and Pix2Pix-based systems for campus and residential layouts [132,133]. Other GAN-based frameworks targeted urban design tasks, such as Urban-GAN [21], MasterplanGAN [134] and models integrating thermal or environmental constraints [135]. These methods demonstrated the feasibility of learning spatial patterns from data but are limited by training instability and difficulty in preserving structural coherence.

Diffusion models provide a more stable and expressive alternative and have been successfully applied to architectural and interior layout generation [75, 136, 137]. Recent studies further enhance spatial realism by

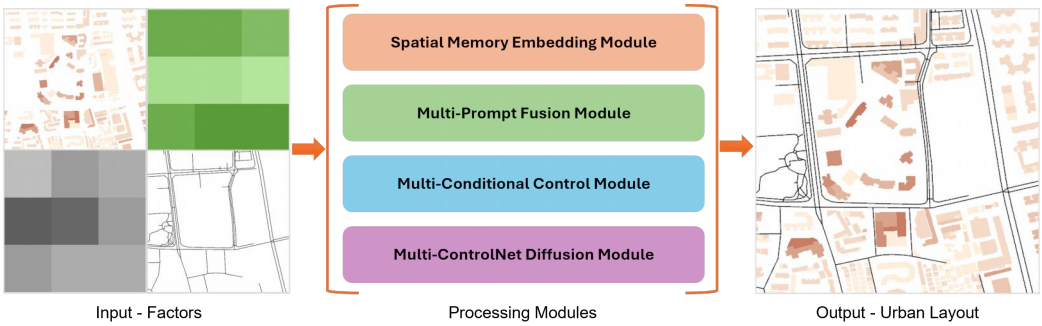


Figure 7.2: Overview of the approach. The model predicts urban layouts by integrating conditional inputs through four core modules: (1) Spatial Memory Embedding; (2) Multi-Prompt Fusion; (3) Multi-Conditional Control; and (4) Multi-ControlNet Diffusion.

incorporating structural priors. GlobalMapper [138] uses graph attention with canonical spatial transformations to generate irregular urban blocks, while CityDreamer [77] and CityGen [78] adopt compositional and semantic diffusion strategies to improve large-scale consistency. Although these diffusion-based methods advance controllability and realism, most remain limited to static layouts and do not address temporal evolution or cross-region coherence.

A second line of research focuses on multi-conditional control. Density maps, height maps and road networks are widely used to guide generation, but many models process these conditions independently and lack mechanisms for semantic fusion [29,139]. Extensions such as House-GAN++ [140], Pix2Pix-based zoning models [61,141] and multi-conditional GANs [142,143] improve controllability but continue to struggle with parameter redundancy and inconsistent performance under varying conditions. Multi-conditional diffusion frameworks such as FloorplanDiffusion [144] increase stability but still rely on multi-stage structures that limit unified semantic integration. These limitations motivate the unified multi-conditional strategy adopted in this chapter.

Modeling long-term spatiotemporal dynamics forms a third research direction. Remote sensing and machine learning techniques have been used to analyze urban morphology, land-use transitions and growth patterns [145–148]. Deep learning models further capture non-linear temporal relationships, including hybrid Random Forest–CNN architectures [149] and temporal convolutional autoencoders for satellite sequences [150]. However, generative AI systems for urban prediction frequently ignore dependencies between neighboring regions and treat patches independently, leading to spa-

tial discontinuity [130] and temporal fragmentation [151]. Graph-based and topology-aware models [152, 153] improve structural reasoning but remain limited to local or short-term relationships.

MMCN builds upon these developments by introducing neighbor-aware spatial memory and temporal consistency mechanisms that explicitly model cross-region and multi-temporal dependencies (Figure 7.2). By integrating multi-conditional control, contextual memory and temporal evolution modeling, MMCN aims to generate urban layouts that are structurally coherent, temporally consistent and better suited for sustainable planning analysis.

The main contributions of this work are summarized as follows:

- A multi-conditional control architecture integrating density, height, transport, and historical layouts for urban evolution modeling.
- A neighbor-aware spatial memory mechanism ensuring cross-region continuity in large-scale urban generation.
- A temporal consistency module capturing long-term evolution trends for sustainable development forecasting.
- A publicly available multi-modal, multi-temporal urban dataset as a benchmark for urban evolution research.

7.2 Methodology

This section introduces the MMCN (Memory-aware Multi-Conditional generation Network) and its core modules. Based on the stable diffusion model, MMCN achieves structured and highly consistent generation for future urban layout forecasting by integrating unified multi-semantic control, memory-aware neighbor perception, and long-term contextual modeling. As shown in Figure 7.3, the overall framework consists of four key components: the semantic fusion module for tasks and conditional prompts, the memory-aware neighbor patches embedding module, the multi-conditional control module, and the diffusion network module with multiple ControlNets.

7.2.1 Overall Framework

Urban layout formation is influenced by the complex interplay of social, economic, environmental, and spatial factors. Among these, spatial structural factors such as building density, building height, and transportation networks play dominant roles in shaping the morphological and functional patterns of cities. In this study, these three spatial indicators, together with the historical building layout, are adopted as the core conditional inputs of the MMCN framework. This choice is motivated by two main considerations.

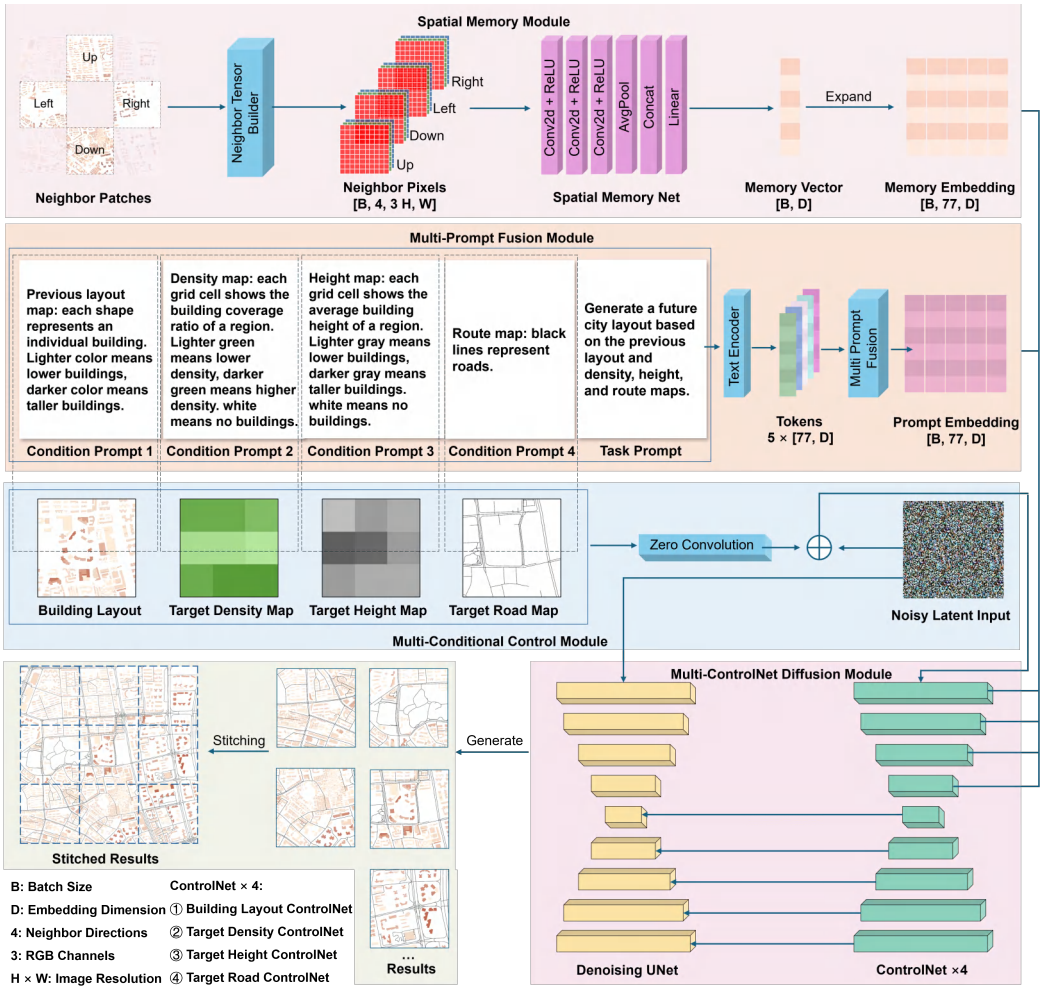


Figure 7.3: Overall framework of MMCN. The model generates future urban layouts via multi-prompt fusion (Sec. 7.2.2), spatial memory embedding (Sec. 7.2.3), multi-conditional control (Sec. 7.2.4), and diffusion-based synthesis with multiple ControlNets (Sec. 7.2.5)

First, these factors represent the most fundamental and universally available indicators across large-scale urban datasets, which facilitates consistent cross-city training and evaluation. Second, they explicitly capture the essential physical and structural characteristics of urban evolution. Building density reflects land use intensity and construction probability, while building height describes the vertical development hierarchy of built environments. Transportation networks determine the accessibility and spatial configuration of urban forms, and the historical layout provides temporal continuity that is crucial for forecasting future development. Compared with socioeconomic

or environmental variables, these spatial layers are more stable, quantifiable, and transferable across different urban contexts, making them particularly suitable for generative modeling of urban layouts.

Building upon these design considerations, the overall architecture of MMCN is constructed to incorporate the selected spatial factors through a multi-conditional control and diffusion-based generation framework. As shown in Figure 7.3, MMCN takes a set of control images, including historical building layout maps, density maps, height maps, road structure maps, and corresponding text prompts for each control image, together with the main task objective prompt as input. Specifically, all prompts are firstly encoded into a shared semantic representation via a text encoder and a multi-head attention-based fusion module. Then, the neighboring images of the current patch (top, bottom, left, right) are processed by a lightweight Convolutional Neural Network (CNN) [154] encoder, and their features are aggregated by a memory embedding module to form a regional context representation. This neighbor image feature representation is added to the shared semantic feature representation to get the final control vector.

The synthesized vector of control features is then embeded into multiple ControlNet branches, each vector is corresponding to a specific type of control condition. The intermediate residuals of all ControlNet branches are aggregated and passed to the U-Net backbone network of the diffusion model for image decoding, generating the target urban layout image for the next time step. During training, the model optimizes both a diffusion-based denoising loss and an edge stitching loss to enhance structural fidelity and ensure natural transitions across patch boundaries. Overall, MMCN provides a robust and scalable solution for temporally consistent urban layout generation through end-to-end semantic-to-spatial unified control and contextual awareness.

7.2.2 Multi-Prompt Fusion Module

In urban evolution prediction tasks, conditional control inputs such as density maps and height maps often represent distinct types of urban information. To effectively integrate control conditions with different semantics, MMCN introduces a multi-prompt semantic fusion mechanism that incorporates these semantics into Multi-Conditional Control Module (Sec. 7.2.4) in a unified and consistent manner.

Specifically, the second section from the top in Figure 7.3 illustrates the Multi-Prompt Fusion Module. In this module, five natural language prompts are designed, including one task-level objective description ("Generate a future city layout based on the previous layout and density, height, and route

maps.") and four specific conditional prompts: density map ("Density map: each grid cell shows the building coverage ratio of a region. Lighter green means lower density, darker green means higher density. white means no buildings."), height map ("Height map: each grid cell shows the average building height of a region. Lighter gray means lower buildings, darker gray means taller buildings. white means no buildings. "), road structure map ("Route map: black lines represent roads."), and historical layout map ("Previous layout map: each shape represents an individual building. Lighter color means lower buildings, darker color means taller buildings").

All prompts are first tokenized and encoded using a pre-trained CLIP text encoder [9], producing token-level embeddings:

$$E_i = \mathrm{Enc}_{\mathrm{CLIP}}(T_i), \quad i = 1, 2, \ldots, N, \tag{7.1}$$

where T_i denotes the i-th text prompt and $E_i \in \mathbb{R}^{B\times L\times D}$ represents its embedding, with B the batch size, L the token length, and D the embedding dimension. All prompt embeddings are then stacked into a 4D tensor $E \in \mathbb{R}^{B\times N\times L\times D}$. In the implementation, $N = 5$ and $L = 77$ are used, following the CLIP tokenizer configuration.

To capture cross-prompt dependencies, MMCN applies a multi-head attention layer to model interactions among the N prompts at each token position. This process can be formally expressed as:

$$F_{\mathrm{mpf}} = \mathrm{Concat}_h(\mathrm{Attn}_h(E))\, W^O, \tag{7.2}$$

where $\mathrm{Attn}_h(E)$ denotes the attention output of the h-th head, $\mathrm{Concat}_h(\cdot)$ indicates concatenation over all H attention heads, and $W^O \in \mathbb{R}^{(HD_h)\times D}$ is a learnable output projection matrix that maps the concatenated features back to the unified embedding dimension D.

The resulting tensor $F_{\mathrm{mpf}} \in \mathbb{R}^{B\times L\times D}$ integrates the semantics of all prompts at the token level, allowing each token to perceive contextual information from other prompts. This unified semantic representation serves as the encoder hidden state shared by all ControlNet branches and forms the text–semantic control input of the MMCN framework.

To further assess the effectiveness of the proposed Multi-Prompt Fusion Module, this work conducted an ablation study to analyze the contribution of each textual prompt. Specifically, this work compared the full configuration using all five prompts with several variants where one or all prompts were removed. The detailed experimental settings and results are provided in Sec. 7.3.5.2.

7.2.3 Spatial Memory Embedding Module

Since directly manipulating the entire city layout is quite challenging, it is common practice to partition the city layout into patches according to specific specifications [73]. The partitioned city layout patches exhibit strong spatial dependencies, such as road continuity. Therefore, ensuring the consistency of patch boundaries is critical to overall generation quality. MMCN introduces a neighbor-aware and memory mechanism, utilizing visual neighbor patches to construct spatial memory embeddings, thereby enhancing continuity and contextual consistency between patches.

The top section of Figure 7.3 shows the Spatial Memory Embedding Module. Specifically, during the generation process, for each target patch, the module dynamically retrieves adjacent images in the up, down, left, and right directions. These neighboring patches are first encoded by a CNN encoder to extract directional visual features, denoted as:

$$F_d = \mathrm{Enc}_{\mathrm{CNN}}(I_d), \quad d \in \{t, b, l, r\}, \tag{7.3}$$

where I_d denotes the neighboring patch in direction d, and $F_d \in \mathbb{R}^{B \times D_f}$ is the corresponding feature vector. t, b, l, and r refer to the top, bottom, left, and right neighboring patches, respectively. If a neighboring patch is missing, a zero vector of the same dimension is used as a placeholder.

The four directional features are concatenated and passed through a fully connected layer FC to produce a unified spatial memory embedding M_{mem}:

$$M_{\mathrm{mem}} = \mathrm{FC}([F_t; F_b; F_l; F_r]) \in \mathbb{R}^{B \times D}. \tag{7.4}$$

To ensure consistency with the token-level prompt semantics, this memory embedding is further expanded to match the shape of the fused semantic tensor:

$$M'_{\mathrm{mem}} = \mathrm{Expand}(M_{\mathrm{mem}}) \in \mathbb{R}^{B \times L \times D}. \tag{7.5}$$

The resulting tensor M'_{mem} provides spatial context information that is later fused with the multi-prompt semantic representation F_{mpf} to form a context-aware control feature for the diffusion model.

7.2.4 Multi-Conditional Control Module

Urban evolution is influenced by various spatial factors, such as building density and height. To fully leverage these diverse control conditions, this work designed a Multi-Conditional Control Module to integrate multiple control signals and effectively guide the image generation process.

Specifically, as shown in the third section from top in Figure 7.3, this Multi-Conditional Control Module introduces four distinct types of control maps: building layout maps, road structure maps, building density maps, and building height maps. Each control map provides key structural or semantic information about the target city's layout. To ensure training stability, each control map is first processed through an independent zero convolution layer. This initialization mechanism ensures that the control maps do not influence the generation process during the early stages of training. As training progresses, the model gradually learns to utilize the conditional information contained in the control maps. The features extracted from the control maps after zero convolution processing are then fused with the noisy latent input during the diffusion process. The fused features are then passed as input to the subsequent Multi-ControlNet Diffusion Module. Through this approach, the control module can simultaneously leverage the structural guidance information provided by the control maps and the generative prior embedded in the noisy latent space.

For each control map C_j (layout, density, height, and route), a structural feature is first extracted via a zero-initialized convolution layer:

$$F_j = \mathrm{Conv}_0(C_j), \qquad j = 1, 2, 3, 4. \tag{7.6}$$

During the diffusion process, these control features are fused with the noisy latent representation z_t to form the conditional input:

$$H_t = \Phi\big(z_t,\, \{F_j\}_{j=1}^{4}\big)\,, \tag{7.7}$$

where $\Phi(\cdot)$ denotes the feature fusion operation that combines the latent and the multi-conditional control features. The resulting representation H_t serves as the input to the subsequent Multi-ControlNet Diffusion Module.

7.2.5 Multi-ControlNet Diffusion Module

Urban evolution is influenced by various spatial factors, including density, height, transportation networks, and historical evolution. To fully utilize these diverse control features, this work adopted a multi-branch ControlNet diffusion architecture in MMCN to perform modeling of multi-modal inputs, while achieving unified control through shared semantic control features.

The bottom right section of Figure 7.3 illustrates the detailed structure of the Multi-ControlNet Diffusion Module. First, all branches share a common encoder hidden state as their initial conditioning input. This hidden state jointly encodes the fused prompt semantics from Multi-Prompt Fusion Module (Sec. 7.2.2) and the spatial neighbor context from Spatial Memory

Embedding Module (Sec. 7.2.3), ensuring consistent guidance across the branches. Second, after passing through Multi-Conditional Control Module (Sec. 7.2.4), the four control maps (density, height, roads, and historical layout) and noisy latent input are fused, and the resulting features are input into independent ControlNet branches. These branches share a common architecture, comprising multiple convolutional layers that extract multi-scale residual features at different downsampling levels.

Subsequently, each control branch guides different stages of the diffusion model independently during the denoising process through residual connections. Specifically, the multi-scale residuals from each ControlNet branch are aggregated at the corresponding layers of the U-Net backbone network, encompassing downstream blocks and intermediate blocks. The outputs from the four branches at each layer are summarized by residual aggregation and used as auxiliary residual input to guide the U-Net decoder for image reconstruction. This architecture enables the generative model to dynamically balance and efficiently integrate multiple conditional constraints at each stage of the denoising process, achieving more precise structural guidance and result control. By integrating multiple ControlNet branches under unified semantic control, MMCN can generate robust structural responses to complex urban environments, enhancing the rationality, consistency, and stability of layouts generated under multi-modal constraints.

Formally, the overall generation process of MMCN can be expressed as:

$$\hat{x}_0 = \Psi_{\text{gen}}(H_t,\, E_{\text{sem}})\,, \tag{7.8}$$

where $\Psi_{\text{gen}}(\cdot)$ denotes the generation function parameterized by the MMCN framework, and $\hat{x}_0$ represents the predicted future urban layout image. The variable H_t is the fused conditional representation obtained from the Multi-Conditional Control Module (see Sec. 7.2.4), and the unified semantic-spatial embedding E_{sem} integrates both textual semantics and spatial context, defined as $E_{\text{sem}} = F_{\text{mpf}} + M'_{\text{mem}}$, where F_{mpf} and M'_{mem} are derived from the Multi-Prompt Fusion Module (Sec. 7.2.2) and the Spatial Memory Embedding Module (Sec. 7.2.3), respectively.

7.2.6 Loss Functions

To optimize the structural accuracy and regional continuity of future urban layout generation, MMCN adopts a joint training objective consisting of two loss components: a denoising loss based on diffusion modeling and an auxiliary edge stitching loss that enhances boundary continuity. This joint objective balances the quality of a single patch generation with smooth transitions between patch boundaries.

7.2.6.1 Denoising Loss

As the core training objective of stable diffusion models, MMCN employs a denoising loss by randomly sampling latent variables with added noise at each training step and guiding the model to predict the original noise distribution. Specifically, at a random timestep $t \in [1, T]$, the model receives a noisy latent image z_t, with the target being the corresponding Gaussian noise ϵ. The denoising loss is computed as the mean squared error (MSE):

$$\mathcal{L}_{\text{den}} = \mathbb{E}_{z,\epsilon,t}\left[\|\epsilon - \hat{\epsilon}_\theta(z_t, t, c)\|_2^2\right] \tag{7.9}$$

where $\hat{\epsilon}_\theta$ is the model's noise prediction, and c is the control vector combining prompt semantics and neighbor context information.

7.2.6.2 Stitching Loss

To enhance structural consistency at patch boundaries, this work introduce an edge stitching loss as an auxiliary objective to penalize inconsistencies between the current patch and its adjacent patches (top, bottom, left, right). This loss encourages spatial consistency by minimizing the discrepancy within the overlapping areas after patch alignment. Specifically, for each valid neighboring direction, this work first stitch the current patch with its corresponding neighbor to form a joint boundary region, and then calculate the L1 loss over this stitched area to measure the remaining discrepancy after alignment. A 100-pixel-wide region is selected along each boundary for this stitching-based evaluation to strike a balance between capturing sufficient contextual information for evaluating spatial continuity and maintaining a localized focus on the most transition-sensitive areas.

$$\mathcal{L}_{\text{st}} = \sum_{d \in \{\text{t,b,l,r}\}} \sum_{i \in \mathcal{R}^d} \left\| S^d(i) - S_{\text{n}}^d(i) \right\|_1 \tag{7.10}$$

where $d \in \{\text{t}, \text{b}, \text{l}, \text{r}\}$ denotes the four directions: top (t), bottom (b), left (l), and right (r). $S^d(i)$ and $S_{\text{n}}^d(i)$ represent the pixel values at position i within the stitched boundary regions of the current patch and its neighboring patch in direction d, respectively. $\mathcal{R}^d$ denotes the set of all pixel positions within the 100-pixel-wide stitched region along direction d. This loss is applied only in directions where valid neighboring patches are available.

7.2.6.3 Joint Loss

The final training objective combines the denoising loss and the edge stitching loss, each with an independent weighting coefficient. The total loss is defined

as:

$$\mathcal{L}_{\text{total}} = \alpha \cdot \mathcal{L}_{\text{den}} + \lambda \cdot \mathcal{L}_{\text{st}} \tag{7.11}$$

where α and λ are the weighting coefficients for the denoising loss $\mathcal{L}_{\text{den}}$ and the edge stitching loss $\mathcal{L}_{\text{st}}$, respectively. These coefficients control the trade-off between reconstruction accuracy and boundary continuity.

In the experiments, this work set $\alpha = 1.0$ to prioritize the denoising objective and set $\lambda = 0.1$ to provide sufficient spatial consistency without overwhelming the primary learning task. This joint loss formulation enables the model to maintain high global structural accuracy while achieving robust local continuity across tile boundaries.

7.3 Experiments

In this section, the details of the experiments are described, including data preparation, qualitative evaluation, quantitative evaluation, and ablation studies, to evaluate the effectiveness of MMCN in predicting urban evolution for sustainable planning.

7.3.1 Implementation Details

All experiments were conducted on a single RTX 5090 GPU with 32 GB of memory. The proposed framework was implemented on top of Stable Diffusion v1.5 (SD1.5), which serves as the generative backbone. It consists of an AutoencoderKL, a UNet2DConditionModel, and four ControlNet branches, further extended by two customized modules—Multi-Prompt Fusion and Spatial Memory Net—to enhance semantic integration and spatial coherence across neighboring regions. All model components were initialized from the pretrained SD1.5 checkpoint to ensure stable convergence.

Training was performed using mixed-precision (FP16) to balance computational efficiency and memory usage. Each input patch was resized to a resolution of 512×512 pixels, with a batch size of 1 and a total of 100,000 optimization steps, corresponding to approximately nine training epochs. The dataset described in Section 7.3.2 provided 22,410 cross-temporal layout samples, of which 80% were used for training and 20% for validation and testing.

During training, GPU memory usage stabilized at around 31.9 GB, with peak utilization reaching about 93%. In inference, generating a single 512×512 layout required roughly 3 seconds and 12.9 GB of GPU memory. These configurations demonstrate that the proposed framework achieves a favorable

balance between model complexity and computational efficiency, and can be efficiently trained and deployed on a single high-memory GPU.

7.3.2 Dataset Preparation

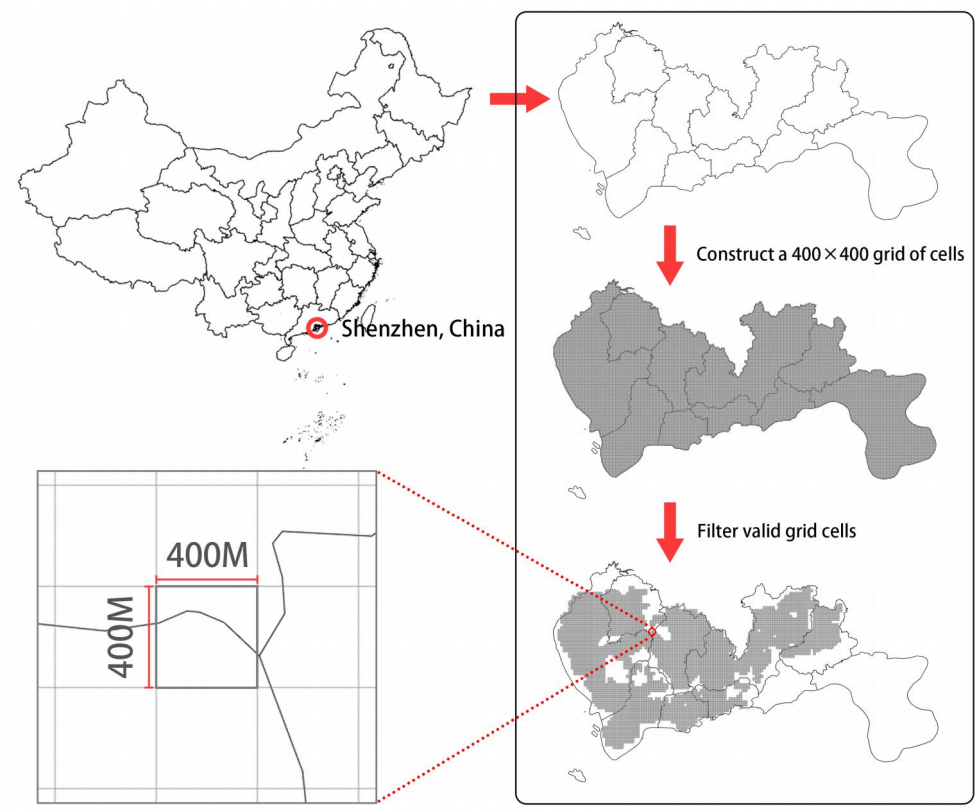


Figure 7.4: Study area and spatial grid sampling process for Shenzhen, China. The left panel shows the geographical location of Shenzhen within China. The right panel illustrates the data preprocessing procedure.

In this paper, a multi-modal, multi-temporal urban layout evolution dataset is constructed for training and validation. Specifically, this work constructed a layout dataset for Shenzhen, China, a rapidly urbanizing city with rich multi-modal and multi-temporal data availability, making it an ideal case for urban evolution modeling. Figure 7.4 illustrates the geographical location of Shenzhen within China and the spatial grid sampling process used to construct the dataset. The city boundary was divided into a uniform grid, and only valid cells containing built-up areas were selected as effective patches for model training and testing. This grid-based sampling

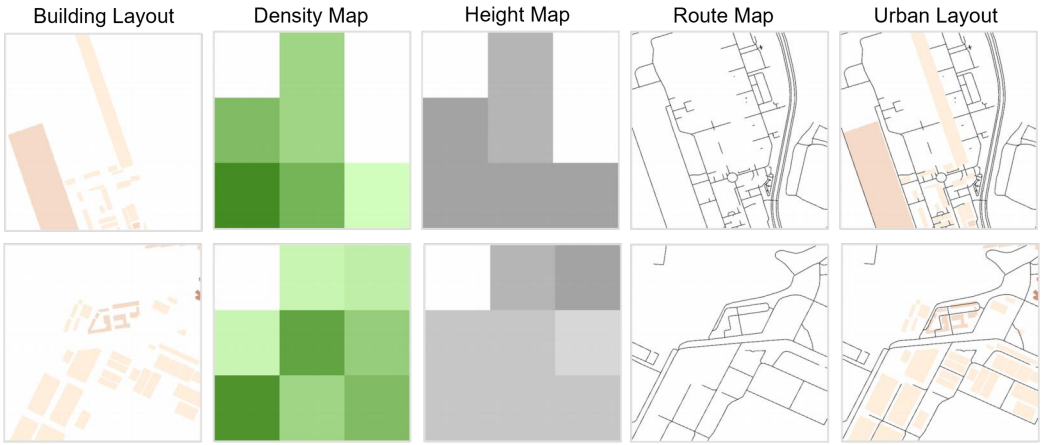


Figure 7.5: Examples of multi-modal inputs and ground truth layouts in the dataset. Each row illustrates a sample consisting of: (1) the historical building layout, (2) the target density map, (3) the target height map, (4) the target road network, and (5) the target urban layout with integrated road and building information.

strategy allows the MMCN framework to operate at a fine spatial resolution while maintaining computational efficiency and scalability across large urban areas. In terms of multi-modal data, the dataset primarily includes five types of data: building layout maps, building height maps, building density maps, route structure maps, and urban layout maps. In terms of multi-temporal data, the dataset primarily covers the years 2005, 2012, 2018, and 2024, spanning nearly two decades of data across four key years for Shenzhen city. The five types of urban data are divided into patches of size 512×512 pixels. The number of layout patches in the dataset is 14,940.

Figure 7.5 shows five types of data examples. The building layout map only includes the shape of buildings, with darker colors indicating taller buildings; building height maps use a 9-grid division, with each grid's color corresponding to the average height of buildings in that area, with darker colors indicating taller buildings; building density maps also use a 9-grid division, with each grid's color corresponding to the ratio of building area to grid area, with darker colors indicating higher ratios; route maps use black lines to indicate road within the patch; The urban layout map is a representation that combines the building layout map and the route map.

During the construction of the training dataset, this work designed cross-temporal training pairs. Specifically, training pairs include the following combinations: 2005 and 2012, 2005 and 2018, 2005 and 2024, 2012 and 2018, 2012 and 2024, and 2018 and 2024. For each temporal pair, the data is

divided into training and validation sets, with 80% used for training and the remaining 20% reserved for validation and testing. This cross-temporal pairing strategy not only increases the volume of training data but also enables the model to learn from multi-interval temporal patterns, thereby enhancing its ability to capture long-term urban evolution trends.

7.3.3 Qualitative Evaluations

To evaluate the performance of the MMCN in urban layout prediction and generation tasks, this work compared it with conventional image translation baselines and a typical multi-conditional controlled diffusion model. Specifically, this work selected Pix2Pix [29], CycleGAN [30], and Instruct-Pix2Pix [155] as baseline methods. Pix2Pix and CycleGAN are representative supervised and unsupervised image-to-image translation frameworks widely used in urban layout synthesis [135, 156], while Instruct-Pix2Pix is a state-of-the-art diffusion-based editing model guided by both structural inputs and textual instructions, making it adaptable for spatial planning tasks [157]. These baselines were chosen to cover both GAN-based translation and diffusion-based generation paradigms, enabling a comprehensive comparison across different generative model families. Quantitative and qualitative comparative experiments were conducted to verify the comprehensive advantages of MMCN in multi-conditional input control, spatial continuity, and semantic consistency.

The generation results of each method are shown in Figure 7.6. Each pair of rows shows the results for a target year corresponding to a different initial historical year. The first column shows the building layout map for the input historical year, and the second column shows the ground truth city layout map. Columns 3 through 6 show the generation results of Pix2Pix, CycleGAN, Instruct-Pix2Pix, and the MMCN, respectively. While the results generated by Pix2Pix can predict some minor changes, they cannot effectively learn the road traffic information of the target layout. The results generated by CycleGAN closely follow the input initial building layout and lack predictive power, failing to effectively generate road information. Furthermore, some of the generated results contain meaningless, noisy data. Instruct-Pix2Pix uses real data as paired data for training, and the generated building layouts align well with the input layout. However, it lacks the ability to predict and evolve future building layout developments. Furthermore, the generated road maps often fill entire areas with blanks, failing to generate effective road plans and differing significantly from real data. In contrast, the generated results can predict how the architectural layout of a city will diverge from its historical layout over time, effectively generating

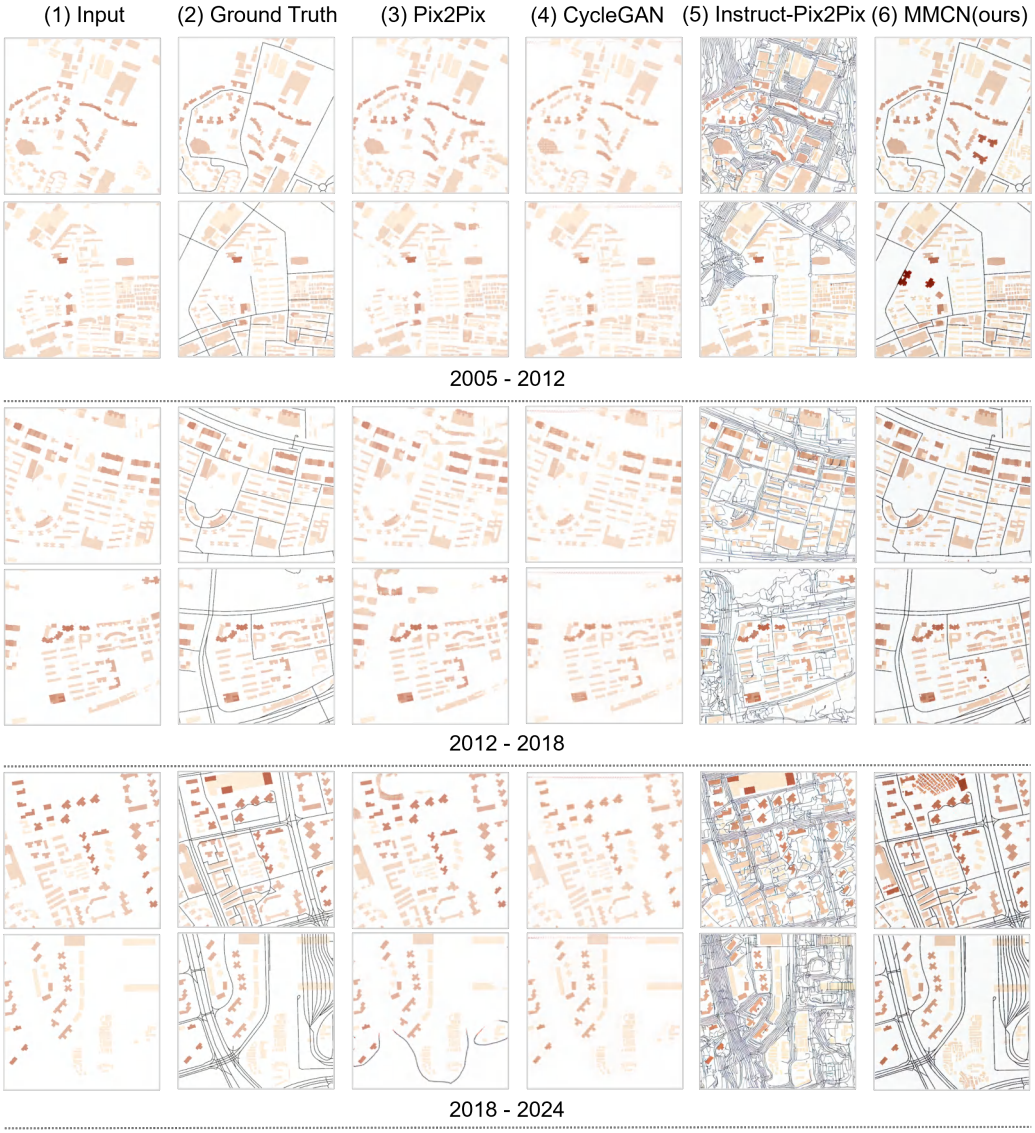


Figure 7.6: Qualitative comparison of urban layout generation results. Each pair of rows corresponds to one test case, showing predictions from different initial historical layouts. From left to right: (1) input historical building layout, (2) ground truth layout for the target year, (3–6) generation results from Pix2Pix [29], CycleGAN [30], Instruct-Pix2Pix [155], and MMCN (ours). Compared to the baselines, MMCN generates layouts with higher structural accuracy, more realistic road patterns, and stronger consistency with the ground truth, demonstrating superior capability in predicting future urban developments.

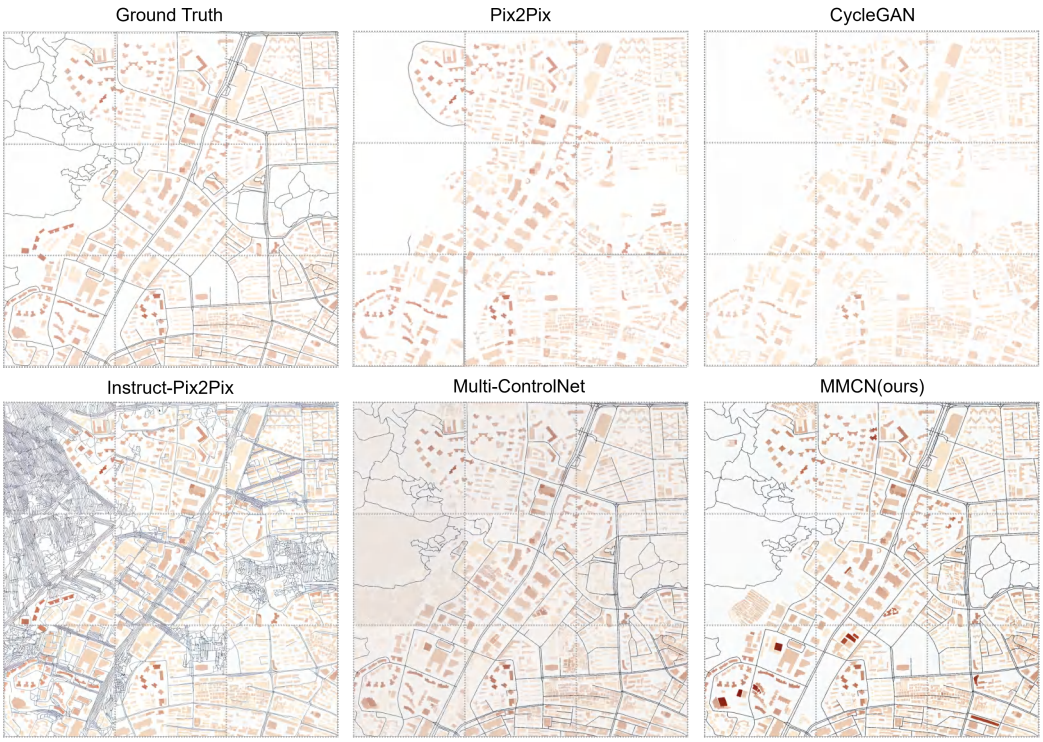


Figure 7.7: Qualitative comparison of stitched layout results based on 9-grid assembly. To evaluate the spatial continuity of generated patches, this work selected 9 adjacent outputs from each method and stitched them into a complete layout. Compared to other methods, MMCN produces better alignment at patch boundaries, preserving road connectivity and building consistency across patches, closely matching the ground truth.

new buildings or deleting existing ones. Furthermore, the generated road structure closely matches the ground truth road traffic. The MMCN model consistently generates city layouts with stronger spatial consistency and relatively more accurate semantic information, demonstrating superior urban layout prediction and generation capabilities.

To validate the ability of generated urban layout patches to be assembled into a complete urban planning layout map, this work conducted verification under a 9-grid assembly rule. As shown in Figure 7.7, this study selected 9 patches generated by various methods and assembled them into a 9-grid image according to the puzzle rules. In Pix2Pix, the generated road networks are sparse and often disconnected when adjacent patches are combined. CycleGAN produces visually continuous patch boundaries but fails to generate effective road structures. Instruct-Pix2Pix generates smoother textures but

still exhibits structural mismatches, such as duplicated or truncated roads along the seams. The Multi-ControlNet model produces more complete road patterns, but it often generates redundant building blocks, which result in uneven boundaries and visual misalignment between patches. Compared to other methods, this model aligns structures better at patch boundaries, effectively integrating building layouts and road routes to achieve a more natural transition, with most of the boundary stitching aligning with the ground truth.

To further highlight the temporal prediction capability of MMCN, the evolution of generated layouts across three consecutive time intervals is additionally visualized, as shown in Figure 7.8. Each row corresponds to a different temporal period (2005–2012, 2012–2018, and 2018–2024), while each column represents a different data source: the input historical layout, the ground truth layout of the target year, and the layout predicted by the MMCN model. The red boxes highlight areas with notable structural changes. These results clearly demonstrate that MMCN can accurately capture both spatial and temporal dynamics of urban evolution, effectively predicting long-term development patterns rather than merely reproducing static layouts.

7.3.4 Quantitative Evaluations

To objectively evaluate the performance of MMCN, this work conducted a comprehensive quantitative assessment of the city layouts generated by different models using four evaluation metrics. First, the Mean Squared Error (MSE) measures pixel-wise differences between the generated layout and the ground-truth layout. Lower MSE values indicate more accurate reconstruction. Second, the Structural Similarity Index (SSIM) evaluates the perceptual similarity and structural coherence of the generated layouts. Higher SSIM values reflect better preservation of overall spatial structures. Third, the Boundary Intersection over Union (Boundary IoU) quantifies the spatial continuity of building boundaries across adjacent layout patches. This metric is particularly important for ensuring seamless transitions between patches; a higher score signifies stronger spatial alignment at patch edges. Lastly, the Planning Alignment Score (PAS) is a domain-specific metric that captures how well the generated layout aligns with key urban planning constraints such as road structures and building locations. A higher PAS indicates better adherence to planning logic and constraints.

As summarized in Table 7.1, MMCN significantly outperforms other baseline models. Specifically, MMCN achieves the lowest MSE and the highest SSIM and PAS scores, demonstrating superior performance in terms

Figure 7.8: Temporal evolution of urban layouts generated by MMCN across three consecutive time intervals (2005–2012, 2012–2018, and 2018–2024).

of both visual fidelity and structural plausibility of the generated layouts. Importantly, MMCN also shows a notable improvement in Boundary IoU, confirming that the model generates more coherent and continuous transitions at tile boundaries. This result highlights the effectiveness of the neighbor-aware context encoding and spatial memory modules in promoting edge consistency and reducing visual artifacts between adjacent patches.

Table 7.1: Quantitative comparison results.

Method	MSE ↓	SSIM ↑	Boundary IoU ↑	PAS ↑
Pix2Pix	547.472	0.772	0.613	2.113
CycleGAN	568.665	0.755	0.538	2.066
Instruct-Pix2Pix	3316.451	0.351	0.376	1.759
Multi-ControlNet	760.923	0.856	0.404	1.796
MMCN (Ours)	**517.851**	**0.885**	**0.642**	**2.419**

7.3.5 Ablation Study

To assess the effectiveness of the proposed MMCN framework, this work conduct two sets of ablation experiments. The first focuses on the overall architectural contributions compared with a strong baseline, while the second examines the role of textual semantics within the Multi-Prompt Fusion module.

7.3.5.1 Comparison with Multi-ControlNet

To assess the effectiveness of the proposed MMCN framework, this work conducted ablation experiments against a strong baseline model, Multi-ControlNet. This baseline maintains the same underlying architecture and training settings but lacks two key contributions: semantic-aware prompt integration and spatial-enhanced neighbor-aware memory embedding.

As shown in Figure 7.9, the Multi-ControlNet model tends to produce discontinuous structures at patch boundaries and often fails to capture high-level urban planning logic, especially when facing complex multi-condition inputs. In contrast, MMCN leverages token-level prompt fusion to enhance semantic consistency and uses neighbor-aware spatial memory to capture long-range dependencies, resulting in more plausible and structurally aligned generations. Figure 7.7 also shows the stitching of the results generated using Multi-ControlNet, which exhibits significant transition discontinuities. As shown in Table 7.1, MMCN consistently outperforms Multi-ControlNet across all evaluation metrics, demonstrating the advantage of incorporating semantic fusion and spatial context modeling. Notably, the improvements in Boundary IoU and PAS are particularly significant, highlighting the effectiveness of the approach in generating spatially coherent and planning-consistent urban layouts. These results confirm that each module in MMCN contributes meaningfully to the overall performance, and their combination is essential for high-fidelity urban layout generation.

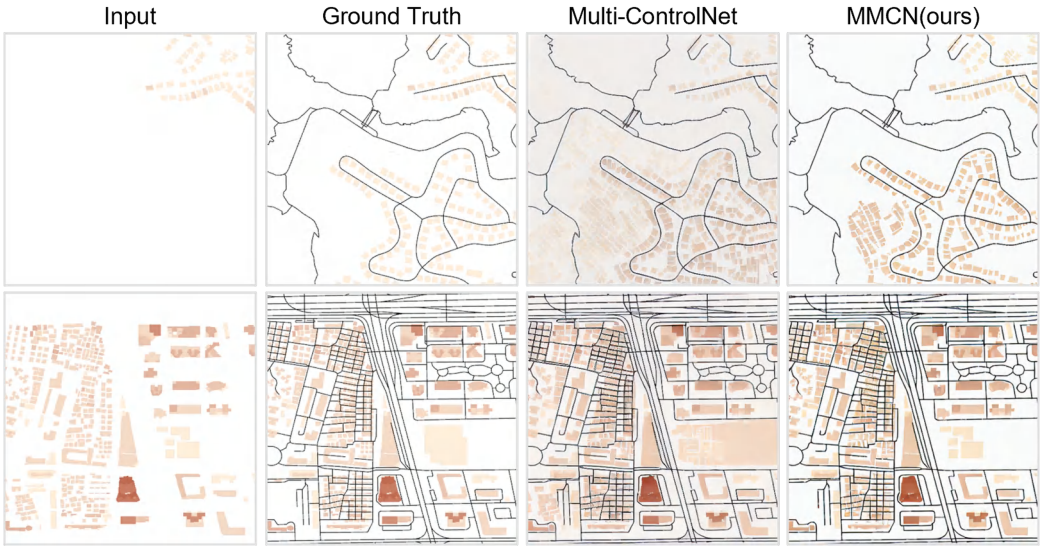


Figure 7.9: Visual comparison of ablation results. MMCN outperforms Multi-ControlNet by generating more coherent and context-aware urban layouts.

7.3.5.2 Multi-Prompt Fusion Analysis

To verify the contribution of each textual prompt, an ablation study was conducted by selectively removing one or all prompts in the Multi-Prompt Fusion Module. Specifically, seven configurations were evaluated as follows: removing the task-level prompt (w/o Task Prompt), the historical layout prompt (w/o Layout Prompt), the density prompt (w/o Density Prompt), the height prompt (w/o Height Prompt), the route prompt (w/o Route Prompt), and the case without any text guidance (No Prompts). The complete setup using all five prompts is denoted as All Prompts.

To quantitatively assess this effect, four complementary metrics are employed to evaluate both structural and perceptual consistency between the generated and ground-truth layouts. The F1 score measures the accuracy of binary structural elements such as building footprints, while the SSIM quantifies structural similarity in terms of local luminance, contrast, and texture. The MSE (Mean Squared Error) captures pixel-level reconstruction error, and the PSNR (Peak Signal-to-Noise Ratio) measures overall signal fidelity in logarithmic decibels, with higher values indicating clearer and more faithful reconstructions.

As shown in Table 7.2, removing any individual prompt slightly decreases the overall performance, indicating that each prompt provides complementary semantic information to the fusion process. The All Prompts configura-

Table 7.2: Effect of removing different textual prompts in the Multi-Prompt Fusion Module.

Configuration	F1 ↑	SSIM ↑	MSE ↓	PSNR ↑
w/o Task Prompt	0.9845	0.854	537.422	20.828
w/o Layout Prompt	0.9842	0.861	526.656	20.916
w/o Density Prompt	0.9845	0.858	542.010	20.791
w/o Height Prompt	0.9834	0.845	571.092	20.564
w/o Route Prompt	0.9855	0.873	519.008	20.979
No Prompts	0.9826	0.835	582.878	20.475
All Prompts (Ours)	**0.9856**	**0.885**	**517.851**	**20.989**

tion consistently achieves the highest scores across all metrics, validating the effectiveness of the proposed token-level semantic fusion. In contrast, the No Prompts configuration yields the lowest performance, confirming that textual semantics, although auxiliary, play a vital role in aligning heterogeneous control inputs (e.g., density, height, and route maps).

7.3.6 Cross-City Generalization Test

To further validate the generalization ability of the proposed MMCN framework, additional experiments were conducted on two other major Chinese cities—Shanghai and Tianjin—which feature distinct urban morphologies and spatial organization patterns compared with Shenzhen. As shown in Figure 7.10 and Figure 7.11, the red boxes highlight areas with notable structural variations, demonstrating that the model maintains stable performance across different urban forms. These results indicate that the proposed approach is not limited to a specific region but can be extended to diverse urban contexts, confirming its broader applicability for general urban layout generation tasks.

7.4 Limitations

Despite the good performance of the proposed MMCN, several limitations remain. As shown in Figure 7.12 (a), three representative failure cases can be observed in the generated urban layouts. In the first row, when the input building layout is completely empty, the expected output should ideally match the ground truth, containing only road structures and leaving the rest blank. However, the model incorrectly generates a large number

Figure 7.10: Qualitative results on the Shanghai dataset.

of meaningless building clusters. This over-generation issue mainly results from the model's learned spatial prior, which tends to generate building-like patterns even in regions without explicit structural cues. During training, the diffusion process focuses on minimizing reconstruction loss across all samples, but regions without explicit negative supervision (i.e., no-building areas) are not effectively represented as empty spaces. As a result, the model interprets these blank inputs as uncertain regions and produces unnecessary building structures instead of maintaining spatial emptiness. In the second row, although the generated layout roughly aligns with the ground truth in terms of spatial regions, it fails to produce a coherent integrated building

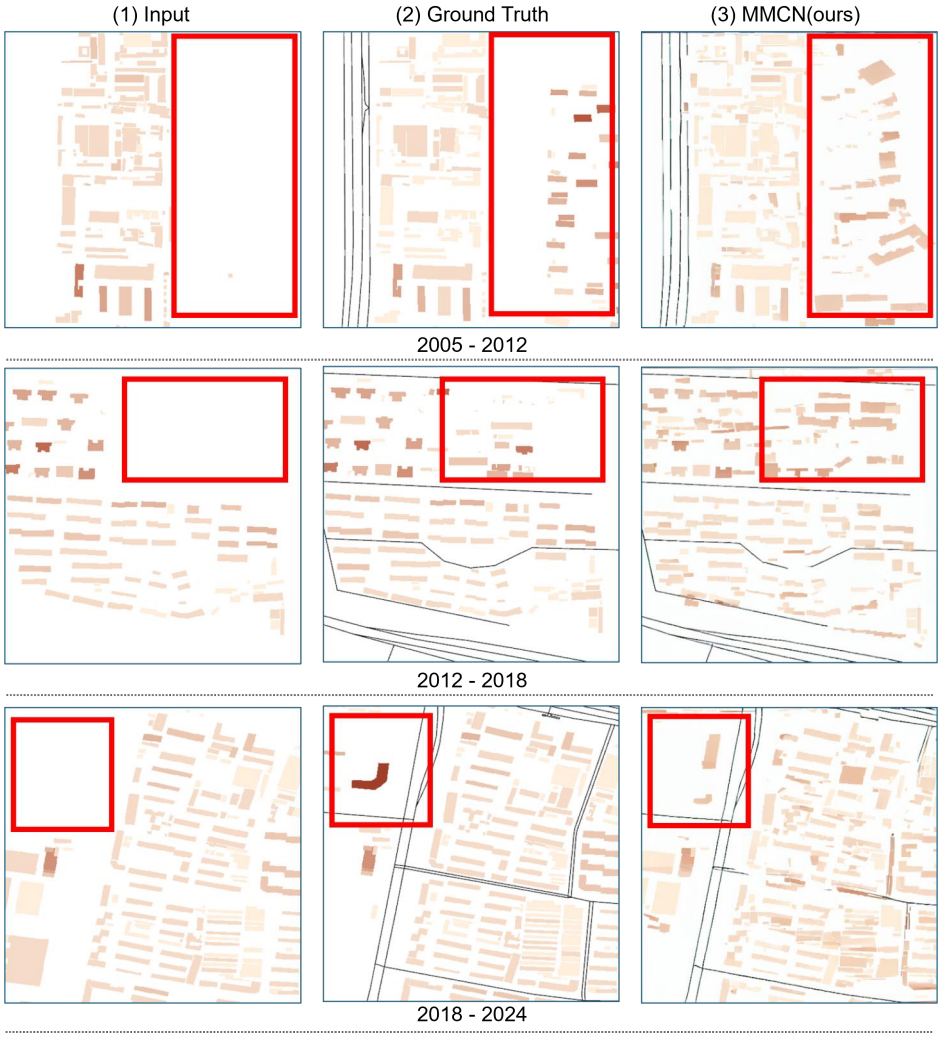


Figure 7.11: Qualitative results on the Tianjin dataset.

structure. Instead, it generates a scattered set of small building clusters that diverge from the intended continuous layout. This issue primarily arises from the open-ended nature of the prediction task. The model is trained to infer future layouts based solely on previous spatial conditions, including layout, density, height, and route maps, without explicit external guidance. In such cases, there is no definitive instruction indicating where new buildings should appear or how they should be organized. As a result, the model attempts to predict plausible structures within uncertain regions, which often leads to fragmented or unrealistic building distributions. If additional control signals such as textual descriptions specifying the number

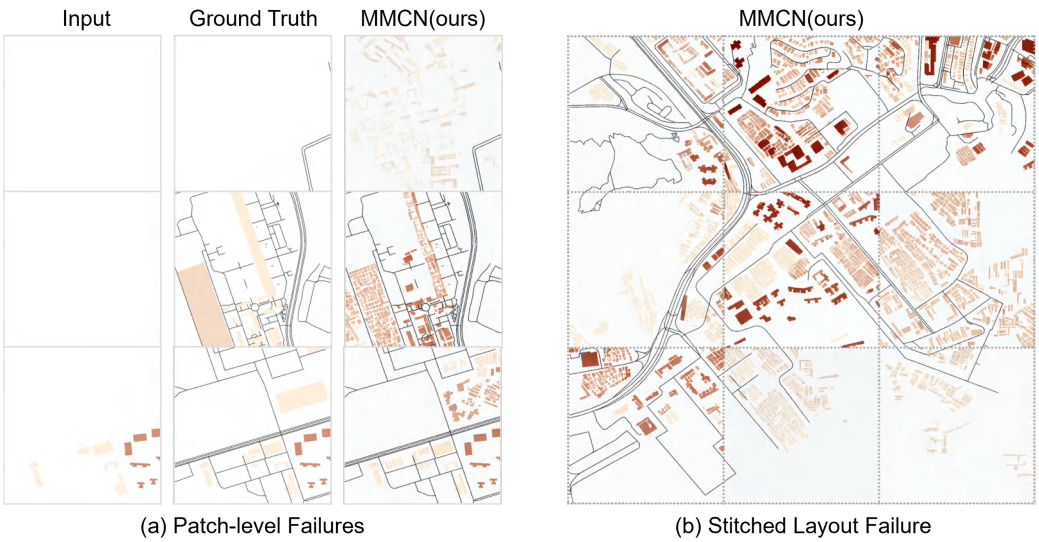


Figure 7.12: Failure cases. (a) Patch-level failures: over-generation in empty inputs, fragmented or incoherent layouts, and unrealistic building placements. (b) Stitched layout composed of these patches, revealing global inconsistencies caused by local failures.

or position of new constructions, or sketch-based guidance indicating desired forms, were provided, the generation process could be guided in a more purposeful and interpretable manner, thereby reducing the occurrence of such fragmented outputs. The third row shows a compound failure case that combines the previous two types: the generated output consists of disjointed, small building fragments, resulting in unrealistic and unstructured layout patterns. When stitching together such low-quality patches, as further illustrated in Figure 7.12 (b), the resulting 3×3 layout grid exhibits clearly visible discontinuities and unnatural transitions at the patch boundaries. These limitations are largely due to the open nature of the generation task. Since the model is designed to predict future urban layouts without strong constraints on the form or function of newly generated buildings, it tends to produce overly freeform outputs, especially in scenarios with minimal semantic input. Future improvements will require mechanisms to better regulate the structure and semantics of the generated layouts.

To address the limitations discussed above, future work will enhance both the controllability and predictive accuracy of the proposed framework. One promising direction is to introduce controllable input conditions that guide the generative process toward more plausible and goal-aligned outcomes. For example, user-defined inputs, such as future building layout sketches,

functional zoning constraints, or region-specific change masks, can serve as auxiliary signals to shape the output in a more purposeful and interpretable manner. These improvements would shift the model from an unconstrained predictor to a scenario-driven planning assistant capable of supporting interactive urban design workflows. In addition, this work intends to expand the dataset to include layout data from a wider range of cities exhibiting diverse urban morphologies and development patterns. This broader data foundation will enable better generalization across different urban contexts and improve the robustness and realism of the generated layouts in real-world scenarios. Through these enhancements, the aim is to build a more practical and reliable generative system for future urban planning, offering strong support for data-driven decision-making processes.

7.5 Conclusion

This paper presented MMCN (Memory-aware Multi-Conditional generation Network), a generative AI-driven framework that advances sustainable urban development planning through multi-factor evolution forecasting. As cities face unprecedented challenges in balancing rapid growth with sustainability objectives, this work provides planners with evidence-based tools to evaluate long-term consequences of development decisions, supporting the achievement of SDG 11 (Sustainable Cities and Communities).

MMCN addresses critical gaps between current AI capabilities and urban planning needs through three key technical contributions. First, the unified multi-conditional control architecture enables planners to model the complex interdependencies between building density, height distributions, transportation networks, and historical patterns. Second, the memory-enhanced spatial mechanism ensures regional coherence essential for metropolitan-scale sustainable development strategies, overcoming the fragmentation that limits existing patch-based methods. Third, the temporal consistency framework leverages historical evolution patterns to enable multi-decade forecasting, allowing planners to assess the cumulative impacts of current decisions on future urban sustainability. A comprehensive evaluation using a novel Shenzhen urban evolution dataset (2005–2024) demonstrates MMCN's effectiveness across multiple dimensions critical to planning practice. The framework achieves superior performance in structural accuracy, spatial continuity, and temporal consistency. In addition to the experiments conducted on Shenzhen, the proposed framework was further evaluated on Shanghai and Tianjin, two cities with distinct urban morphologies and development patterns. The consistent performance across these cities demonstrates that

MMCN is not overfitted to a single urban context, but exhibits strong cross-city generalization capability. These results indicate that the framework can be extended to diverse metropolitan environments, supporting broader applicability in real-world urban planning scenarios.

Future research may focus on enhancing the effectiveness of MMCN in supporting sustainable urban planning. Integration with climate models could enable the assessment of development scenarios' environmental impacts. Incorporation of socio-economic factors would support more comprehensive sustainability analysis. Development of interactive interfaces would facilitate participatory planning processes, allowing stakeholders to collaborate in shaping sustainable urban futures. Additionally, expanding the framework to handle diverse urban morphologies globally would increase its applicability to different planning contexts and cultural settings.

In conclusion, MMCN represents a significant step toward AI-assisted sustainable urban planning, demonstrating how advanced generative models can be purposefully designed to address real-world sustainability challenges. As urbanization accelerates globally, tools that enable evidence-based, long-term planning become increasingly critical. By bridging the gap between AI innovation and planning practice, this work contributes to the broader goal of creating sustainable, resilient, and livable cities for future generations. It is hoped that MMCN will inspire further research at the intersection of AI and urban sustainability, ultimately supporting the transformation of the cities toward more sustainable development pathways.

Chapter 8

Conclusion

This dissertation has presented a unified cross-level generative modeling approach that integrates three domains of the built environment traditionally studied in isolation: object-level 3D modeling, architectural form generation, and urban-level spatial evolution forecasting. Although recent advances in generative artificial intelligence have significantly improved multimodal reasoning and visual synthesis, their application to architectural and urban tasks has been constrained by a lack of structural understanding, geometric consistency, and temporal continuity. The research conducted in this dissertation addresses these limitations through a set of models, datasets, and algorithms that enable generative AI to operate as a spatial reasoning system across multiple levels, with each level deliberately designed to inform and constrain the next.

At the form and object level, the DualShape module demonstrated that freehand sketches can serve as effective inputs for component-aware 3D generation and retrieval. By combining implicit shape representations with explicit geometric decomposition, this module provides a foundation for semantic and part-based reasoning that can be extended to architectural and urban contexts, where similar part–whole relationships recur at larger spatial scales.

At the architectural level, the dissertation extends object-level geometric understanding to buildings as a specific and structured design object, introducing three contributions that collectively enhance structural coherence and representational consistency. First, a large-scale multi-LoD sketch dataset was created through a deterministic–generative pipeline, enabling models to learn hierarchical abstraction from detailed architectural geometry. Second, a multi-view generation framework was proposed to produce geometry-consistent architectural renderings from massing models. This framework integrates multi-view ControlNet, depth-informed geometric priors, and image-space consistency mechanisms to maintain alignment across viewpoints. Third, a VLM-guided facade renovation workflow was developed, combining semantic reasoning, component-level generation, and photorealistic refinement to support early-stage adaptive reuse scenarios. These architectural-

level methods demonstrate how generative models can interpret, abstract, and restructure architectural geometry rather than simply producing visually plausible images.

At the urban level, the Memory-Aware Multi-Conditional Generation Network (MMCN) extends building-level constrained generation to city-scale spatial evolution, where consistency, continuity, and temporal dependency become dominant concerns. The framework unifies heterogeneous spatial factors—including historical layouts, density distributions, height patterns, and transportation networks—within a multi-conditional diffusion process. A multi-prompt semantic fusion mechanism ensures coherent alignment between textual and visual conditions, while a spatial memory embedding module enhances continuity across adjacent urban patches. Experiments on Shenzhen's multi-temporal dataset (2005–2024) indicate that MMCN achieves improvements in structural accuracy, boundary continuity, and temporal consistency. Additional evaluations on Shanghai and Tianjin confirm the generalization capability of the proposed approach.

Beyond empirical findings, this dissertation contributes conceptually and methodologically to generative modeling in the built environment. It formulates cross-level generative modeling as a core methodological contribution, linking object-level form representation, building-level spatial reasoning, and urban-level dynamics within a unified computational paradigm. It further formalizes multi-LoD sketches and architectural abstractions as structured representations that encode both geometric and semantic information. Finally, it proposes consistency mechanisms—such as token-level semantic fusion, neighbor-aware memory embedding, multi-conditional ControlNet modeling, and multi-view image-space coherence—that enable generative models to produce spatially aligned and semantically meaningful results across levels.

Taken as a whole, the research presented here demonstrates that generative AI has the potential to evolve from a visualization-centric technique into a comprehensive approach for spatial reasoning, capable of supporting tasks ranging from conceptual form exploration to long-term urban planning. The results suggest a trajectory in which generative AI becomes more deeply integrated into the processes through which the built environment is designed, analyzed, and projected over time.

8.1 Limitations

Although the proposed framework demonstrates strong performance across object-, building-, and city-level tasks, several limitations still remain.

At the object level, the part-aware 3D generation module depends on manually defined assembly priors. This reliance makes it difficult to extend the system to a wide range of categories. The method is also sensitive to sketches that contain very limited or ambiguous structural cues, which can lead to unreliable part retrieval or incorrect geometric inference. In addition, the hybrid implicit–explicit representation used for generation may produce inconsistencies in mesh resolution and surface quality.

At the building level, the LoD sketch abstraction method focuses primarily on geometric simplification and does not yet incorporate BIM semantics, material attributes, or structural annotations. As a result, the extracted multi-LoD representations have limited applicability in downstream engineering or simulation workflows. The multi-view generation framework is based on simplified shoebox massing models. These abstractions cannot fully describe complex architectural elements such as internal voids, stair cores, or intricate facade structures, and this occasionally leads to a mismatch between generated images and the actual architectural geometry. The facade renovation workflow is restricted by the characteristics of its training data, which largely consists of simplified, front-facing sketches. Furthermore, the renovation model does not account for structural or regulatory constraints, and therefore may produce solutions that are visually realistic but not feasible in practice.

At the urban level, MMCN may generate building clusters in regions that should remain empty because the model tends to follow its learned spatial priors instead of preserving blank areas. In regions with insufficient semantic cues, the model can produce fragmented or weakly organized building structures. When multiple patches are assembled to form large urban areas, inconsistencies may appear at patch boundaries. More broadly, the generative process remains relatively unconstrained and may yield plausible yet semantically ambiguous future layouts.

Across all spatial levels, a conceptual limitation also remains. Although the proposed components share several methodological principles, they do not yet constitute a fully unified cross-level representation. A coherent connection between object-level geometry, building-level components, and urban spatial structures has not been completely established within a single semantic or coordinate framework.

8.2 Remaining Challenges and Future Work

Despite the breadth of contributions, several limitations remain and point toward promising directions for future research.

First, although this dissertation introduced multiple datasets for LoD abstraction, multi-view generation, industrial renovation, and urban evolution, these datasets are geographically and typologically limited. Expanding data coverage to include diverse climatic, cultural, and regulatory contexts would improve the robustness and applicability of future generative models. Integrating BIM semantics, material properties, and indoor–outdoor spatial structures would further enhance their capacity for structural reasoning.

Second, the generative models developed in this research primarily optimize for perceptual and geometric fidelity. They do not yet incorporate engineering constraints such as structural performance, constructability, environmental metrics, or code compliance. A natural extension is to couple generative models with simulation-based evaluations, such as daylight analysis, energy modeling, or microclimate simulations, thereby enabling performance-aware generative design.

Third, while the proposed models enhance multi-view, multi-LoD, and multi-conditional consistency, finer-grained control remains a challenge. Designers often require explicit manipulation of spatial programs, material systems, facade articulation, or phased development strategies. Modular architectures, constraint-based diffusion models, and interactive design interfaces may offer pathways toward more controllable and interpretable generation.

Fourth, although MMCN captures multi-temporal spatial patterns, urban evolution is influenced by socio-economic processes, planning policies, market forces, and unforeseen environmental events that extend beyond purely spatial data. Combining generative models with agent-based simulations, econometric forecasting, or regulatory datasets would enable more comprehensive modeling of urban dynamics and planning scenarios.

Finally, a longer-term research direction involves extending the components developed in this dissertation toward autonomous AI design agents capable of multi-step reasoning, iterative exploration, and integration with digital twin platforms or robotic fabrication systems. Such agents could support real-time decision-making across levels, from detailed architectural modifications to city-level scenario planning.

In summary, the work presented in this dissertation provides a foundation for cross-level generative modeling of the built environment. Future research may deepen structural integration, broaden data diversity, incorporate performance-driven constraints, and enable greater controllability and interpretability. These directions hold the potential to advance generative AI into a robust and practical methodology for sustainable architectural design and urban planning, shaping more intelligent and adaptive design ecosystems in the years to come.

References

[1] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes, *Computer Graphics: Principles and Practice*. Reading, MA: Addison-Wesley, 1990.

[2] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 25, 2012.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Neural Information Processing Systems (NeurIPS)*, 2014.

[6] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[7] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.

[8] L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text-to-image diffusion models," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 3836–3847.

[9] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PmLR, 2021, pp. 8748–8763.

[10] OpenAI, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[11] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[12] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *International Conference on Learning Representations (ICLR)*, 2014.

[13] C. Eastman, P. Teicholz, R. Sacks, and K. Liston, *BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors*. New Jersey: John Wiley & Sons, 2011.

[14] R. Woodbury, *Elements of Parametric Design*. Routledge, 2010.

[15] F. Biljecki, H. Ledoux, and J. Stoter, "Formalisation of the level of detail in 3d city modelling," *Computers, Environment and Urban Systems*, vol. 48, pp. 1–15, 2014.

[16] S. Chaillou, "Ai and architecture: Towards a new approach," Harvard GSD / Medium, 2020.

[17] Y. Yu, C. Ye, and Y. Wang, "Ai-assisted generative design in architecture," *Frontiers of Architectural Research*, vol. 11, no. 6, pp. 1250–1265, 2022.

[18] UN-Habitat, "World cities report 2022: Envisaging the future of cities," https://unhabitat.org/wcr/, 2022.

[19] W. J. Mitchell, *E-topia: Urban Life, Jim—But Not As We Know It*. MIT Press, 2021, reprint Edition.

[20] S. Fedorova, "Gans for urban design," *arXiv preprint arXiv:2105.01727*, 2021.

[21] S. J. Quan, "Urban-gan: An artificial intelligence-aided computation system for plural urban design," *Environment and Planning B: Urban Analytics and City Science*, vol. 49, no. 9, p. 2500–2515, 2022.

[22] X. Du, T. Zhang, and H. Xie, "Dualshape: Sketch-based 3d shape design with part generation and retrieval," *IEEE Access*, vol. 12, pp. 18 888–18 900, 2024.

[23] X. Du, A. Kongkaeo, Y. Zhang, and H. Xie, "Automatic lod sketch extraction from architectural models using generative ai: Dataset construction for multi-level architectural design generation," in *Proceedings of CAADRIA 2026*, Hsinchu, Taiwan, Apr. 2026.

[24] X. Du, R. Gui, Z. Wang, Y. Zhang, and H. Xie, "Multi-view depth consistent image generation using generative ai models: Application on architectural design of university buildings," in *Proceedings of CAADRIA 2025*, Tokyo, Japan, Mar. 2025.

[25] W. Booranamaitree, X. Du, Y. Cai, Z. Wang, Y. Zhang, and H. Xie, "Sketch-based facade renovation with generative ai: A streamlined framework for bypassing as-built modeling in industrial adaptive reuse," in *Proceedings of CAADRIA 2026*, Hsinchu, Taiwan, Apr. 2026.

[26] X. Du, C. Li, Q. Li, Y. Lu, Y. Xu, Y. Zhang, Z. Xu, and H. Xie, "Ai-driven urban evolution forecasting: A unified memory-aware multi-conditional generation framework for sustainable development planning," *Sustainable Cities and Society*, 2025.

[27] S. Xie and Z. Tu, "Holistically-nested edge detection," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1395–1403.

[28] Y. Liu, M.-M. Cheng, X. Hu, K. Wang, and X. Bai, "Richer convolutional features for edge detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3000–3009.

[29] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[30] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.

[31] T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin, and D. Jacobs, "A search engine for 3d models," *ACM Transactions on Graphics (TOG)*, vol. 22, no. 1, pp. 83–105, 2003.

[32] D.-Y. Chen, X.-P. Tian, Y.-T. Shen, and M. Ouhyoung, "On visual similarity based 3d model retrieval," in *Computer graphics forum*, vol. 22, no. 3. Wiley Online Library, 2003, pp. 223–232.

[33] M. Eitz, R. Richter, T. Boubekeur, K. Hildebrand, and M. Alexa, "Sketch-based shape retrieval," *ACM Transactions on graphics (TOG)*, vol. 31, no. 4, pp. 1–10, 2012.

[34] Q. Yu, Y. Yang, F. Liu, Y.-Z. Song, T. Xiang, and T. M. Hospedales, "Sketch-a-net: A deep neural network that beats humans," *International journal of computer vision*, vol. 122, no. 3, pp. 411–425, 2017.

[35] A. Qi, Y. Gryaditskaya, J. Song, Y. Yang, Y. Qi, T. M. Hospedales, T. Xiang, and Y.-Z. Song, "Toward fine-grained sketch-based 3d shape retrieval," *IEEE Transactions on Image Processing*, vol. 30, pp. 8595–8606, 2021.

[36] K. Mo, S. Zhu, A. X. Chang, L. Yi, S. Tripathi, L. J. Guibas, and H. Su, "Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 909–918.

[37] T. Igarashi, S. Matsuoka, and H. Tanaka, "Teddy: A sketching interface for 3d freeform design," in *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '99. USA: ACM Press/Addison-Wesley Publishing Co., 1999, p. 409–416. [Online]. Available: https://doi.org/10.1145/311535.311602

[38] O. A. Karpenko and J. F. Hughes, "Smoothsketch: 3d free-form shapes from complex sketches," in *ACM SIGGRAPH 2006 Papers*, 2006, pp. 589–598.

[39] A. Nealen, T. Igarashi, O. Sorkine, and M. Alexa, "Fibermesh: designing freeform surfaces with 3d curves," in *ACM SIGGRAPH 2007 papers*, 2007, pp. 41–es.

[40] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese, "3d-r2n2: A unified approach for single and multi-view 3d object reconstruction," in *European conference on computer vision*. Springer, 2016, pp. 628–644.

[41] J. Delanoy, M. Aubry, P. Isola, A. A. Efros, and A. Bousseau, "3d sketching using multi-view deep volumetric prediction," *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, vol. 1, no. 1, pp. 1–22, 2018.

[42] Z. Lun, M. Gadelha, E. Kalogerakis, S. Maji, and R. Wang, "3d shape reconstruction from sketches via multi-view convolutional networks," in *2017 International Conference on 3D Vision (3DV)*. IEEE, 2017, pp. 67–77.

[43] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "Deepsdf: Learning continuous signed distance functions for shape representation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 165–174.

[44] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger, "Occupancy networks: Learning 3d reconstruction in function space," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4460–4470.

[45] J. Li, K. Xu, S. Chaudhuri, E. Yumer, H. Zhang, and L. Guibas, "Grass: Generative recursive autoencoders for shape structures," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, pp. 1–14, 2017.

[46] K. Mo, P. Guerrero, L. Yi, H. Su, P. Wonka, N. Mitra, and L. Guibas, "Structurenet: Hierarchical graph networks for 3d shape generation," *ACM Transactions on Graphics (TOG), Siggraph Asia 2019*, vol. 38, no. 6, p. Article 242, 2019.

[47] C. Li, T. Zhang, X. Du, Y. Zhang, and H. Xie, "Generative ai models for different steps in architectural design: A literature review," *Frontiers of Architectural Research*, 2024.

[48] H. Kato, Y. Ushiku, and T. Harada, "Neural 3d mesh renderer," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3907–3916.

[49] T. Groueix, M. Fisher, V. G. Kim, B. Russell, and M. Aubry, "AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation," in *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[50] M. Tatarchenko, A. Dosovitskiy, and T. Brox, "Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2088–2096.

[51] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 652–660.

[52] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017.

[53] H. Fan, H. Su, and L. J. Guibas, "A point set generation network for 3d object reconstruction from a single image," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 605–613.

[54] G. Gröger and L. Plümer, "Citygml–interoperable semantic 3d city models," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 71, pp. 12–33, 2012.

[55] M. Niemeyer, L. Mescheder, M. Oechsle, and A. Geiger, "Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 3501–3512.

[56] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm," *ACM siggraph computer graphics*, vol. 21, no. 4, pp. 163–169, 1987.

[57] E. Remelli, A. Lukoianov, S. Richter, B. Guillard, T. Bagautdinov, P. Baque, and P. Fua, "Meshsdf: Differentiable iso-surface extraction," *Advances in Neural Information Processing Systems*, vol. 33, pp. 22 468–22 478, 2020.

[58] C. Niu, J. Li, and K. Xu, "Im2struct: Recovering 3d shape structure from a single rgb image," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4521–4529.

[59] F. Biljecki, H. Ledoux, and J. Stoter, "An improved lod specification for 3d building models," *Computers, environment and urban systems*, vol. 59, pp. 25–37, 2016.

[60] Z. Luo and W. Huang, "Floorplangan: Vector residential floorplan adversarial generation," *Automation in Construction*, vol. 142, p. 104470, 2022.

[61] S. Chaillou, "Archigan: Artificial intelligence x architecture," in *Architectural intelligence: Selected papers from the 1st international conference on computational design and robotic fabrication (CDRF 2019)*. Springer, 2020, pp. 117–127.

[62] C. Sun, Y. Zhou, and Y. Han, "Automatic generation of architecture facade for historical urban renovation using generative adversarial network," *Building and Environment*, vol. 212, p. 108781, 2022.

[63] D. Rezende and S. Mohamed, "Variational inference with normalizing flows," in *International conference on machine learning*. PMLR, 2015, pp. 1530–1538.

[64] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=PxTIG12RRHS

[65] A. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew, I. Sutskever, and M. Chen, "Glide: Towards photorealistic image generation and editing with text-guided diffusion models," *arXiv preprint arXiv:2112.10741*, 2021.

[66] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv:2204.06125*, vol. 1, no. 2, p. 3, 2022.

[67] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans *et al.*, "Photorealistic text-to-image diffusion models with deep language understanding," *Advances in neural information processing systems*, vol. 35, pp. 36 479–36 494, 2022.

[68] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in neural information processing systems*, vol. 34, pp. 8780–8794, 2021.

[69] J. Ho and T. Salimans, "Classifier-free diffusion guidance," *arXiv preprint arXiv:2207.12598*, 2022.

[70] C. Mou, X. Wang, L. Xie, Y. Wu, J. Zhang, Z. Qi, and Y. Shan, "T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 38, no. 5, 2024, pp. 4296–4304.

[71] N. Liu, S. Li, Y. Du, A. Torralba, and J. B. Tenenbaum, "Compositional visual generation with composable diffusion models," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv,*

Israel, October 23–27, 2022, Proceedings, Part XVII. Springer, 2022, pp. 423–439.

[72] N. Nauata, K.-H. Chang, C.-Y. Cheng, G. Mori, and Y. Furukawa, "House-gan: Relational generative adversarial networks for graph-constrained house layout generation," in *European Conference on Computer Vision*. Springer, 2020, pp. 162–177.

[73] L. Xu, Y. Xiangli, A. Rao, N. Zhao, B. Dai, Z. Liu, and D. Lin, "Blockplanner: City block generation with vectorized graph representation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5077–5086.

[74] M. Johanes and J. Huang, "Generative isovist transformer: machine learning for spatial sequence synthesis," in *41st Conference on Education and Research in Computer Aided Architectural Design in Europe, eCAADe 2023*. Education and research in Computer Aided Architectural Design in Europe, 2023, pp. 471–480.

[75] M. A. Shabani, S. Hosseini, and Y. Furukawa, "Housediffusion: Vector floorplan generation via a diffusion model with discrete and continuous denoising," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 5466–5475.

[76] H. Ye, J. Zhang, S. Liu, X. Han, and W. Yang, "Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models," *arXiv preprint arXiv:2308.06721*, 2023.

[77] H. Xie, Z. Chen, F. Hong, and Z. Liu, " CityDreamer: Compositional Generative Model of Unbounded 3D Cities ," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, Jun. 2024, pp. 9666–9675.

[78] J. Deng, W. Chai, J. Guo, Q. Huang, J. Huang, W. Hu, S. Hao, J.-N. Hwang, and G. Wang, "Citygen: Infinite and controllable city layout generation," 2025.

[79] G. Fahim, K. Amin, and S. Zarif, "Single-view 3d reconstruction: A survey of deep learning methods," *Computers & Graphics*, vol. 94, pp. 164–190, 2021.

[80] J. A. Sethian, *Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science*. Cambridge university press, 1999, vol. 3.

[81] C. Ding and L. Liu, "A survey of sketch based modeling systems," *Frontiers of Computer Science*, vol. 10, no. 6, pp. 985–999, 2016.

[82] J. Loffler, "Content-based retrieval of 3d models in distributed web databases by visual shape information," in *2000 IEEE Conference on Information Visualization. An International Conference on Computer Visualization and Graphics*. IEEE, 2000, pp. 82–87.

[83] C. Ma, X. Yang, C. Zhang, X. Ruan, and M.-H. Yang, "Sketch retrieval via local dense stroke features," *Image and Vision Computing*, vol. 46, pp. 64–73, 2016.

[84] C. Li, H. Pan, Y. Liu, X. Tong, A. Sheffer, and W. Wang, "Robust flow-guided neural prediction for sketch-based freeform surface modeling," *ACM Trans. Graph.*, vol. 37, no. 6, dec 2018.

[85] X. Han, C. Gao, and Y. Yu, "Deepsketch2face: A deep learning based sketching system for 3d face and caricature modeling," *ACM Trans. Graph.*, vol. 36, no. 4, jul 2017.

[86] G. Nishida, I. Garcia-Dorado, D. G. Aliaga, B. Benes, and A. Bousseau, "Interactive sketching of urban procedural models," *ACM Trans. Graph.*, vol. 35, no. 4, jul 2016.

[87] J. Li, C. Niu, and K. Xu, "Learning part generation and assembly for structure-aware shape synthesis," *CoRR*, vol. abs/1906.06693, 2019.

[88] D. Du, H. Zhu, Y. Nie, X. Han, S. Cui, Y. Yu, and L. Liu, "Learning part generation and assembly for sketching man-made objects," *Computer Graphics Forum*, vol. 40, no. 1, pp. 222–233, 2021.

[89] T. Igarashi and J. F. Hughes, "A suggestive interface for 3d drawing," *ACM SIGGRAPH 2007 courses*, 2001.

[90] Y. J. Lee, C. L. Zitnick, and M. F. Cohen, "Shadowdraw: Real-time user guidance for freehand drawing," *ACM Trans. Graph.*, vol. 30, no. 4, jul 2011.

[91] Z. Huang, Y. Peng, T. Hibino, C. Zhao, H. Xie, T. Fukusato, and K. Miyata, "dualface: Two-stage drawing guidance for freehand portrait sketching," *Computational Visual Media*, vol. 8, pp. 63–77, 2022.

[92] Z. Huang, H. Xie, T. Fukusato, and K. Miyata, "Anifacedrawing: Anime portrait exploration during your sketching," in *ACM*

SIGGRAPH 2023 Conference Proceedings, ser. SIGGRAPH '23. New York, NY, USA: Association for Computing Machinery, 2023. [Online]. Available: https://doi.org/10.1145/3588432.3591548

[93] Y. Peng, Z. Huang, C. Zhao, H. Xie, T. Fukusato, and K. Miyata, "Sketch-based human motion retrieval via shadow guidance," in *2021 Nicograph International (NicoInt)*, 2021, pp. 42–45.

[94] A. Limpaecher, N. Feltman, A. Treuille, and M. Cohen, "Real-time drawing assistance through crowdsourcing," *ACM Trans. Graph.*, vol. 32, no. 4, jul 2013.

[95] E. Iarussi, A. Bousseau, and T. Tsandilas, "The drawing assistant: Automated drawing guidance and feedback from photographs," in *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, ser. UIST '13. New York, NY, USA: Association for Computing Machinery, 2013, p. 183–192.

[96] E. Kalogerakis, M. Averkiou, S. Maji, and S. Chaudhuri, "3d shape segmentation with projective convolutional networks," in *proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3779–3788.

[97] D. Zhang, "Opensse: Open sketch search engine," https://github.com/zddhub/opensse, 2017.

[98] X. Du, Y. He, X. Yang, C.-M. Chang, and H. Xie, "Sketch-based 3d shape modeling from sparse point clouds," in *International Workshop on Advanced Imaging Technology (IWAIT) 2022*, vol. 12177. SPIE, 2022, pp. 714–719.

[99] B. Guillard, E. Remelli, P. Yvernay, and P. Fua, "Sketch2mesh: Reconstructing and editing 3d shapes from sketches," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 13 023–13 032.

[100] P. Li, B. Li, and Z. Li, "Sketch-to-architecture: Generative ai-aided architectural design," in *Proceedings of the 31st Pacific Conference on Computer Graphics and Applications*. The Eurographics Association, 2023.

[101] Open Geospatial Consortium (OGC), "Ogc city geography markup language (citygml) encoding standard, version 2.0.0," Open Geospatial Consortium, Tech. Rep. 12-019, 2012, technical Report.

[102] F. Leite, A. Akcamete, B. Akinci, G. Atasoy, and S. Kiziltas, "Analysis of modeling effort and impact of different levels of detail in building information models," *Automation in Construction*, vol. 20, no. 5, pp. 601–609, 2011.

[103] A. Stadler and T. H. Kolbe, "Spatio-semantic coherence in the integration of 3d city models," in *ISPRS International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 36, 2012, p. 8.

[104] T. Kutzner, K. Chaturvedi, and T. H. Kolbe, "Citygml 3.0: New functions open up new applications," *PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, vol. 88, pp. 43–61, 2020.

[105] S. Dong, "Research on intelligent generation of line drawings from built heritage images," *Buildings*, vol. 15, no. 18, p. 3341, 2025.

[106] R. Zhang, S. Pan, C. Lv, M. Gong, and H. Huang, "Architectural colod generation: Controllable level-of-detail generation for architectural modeling," *ACM Transactions on Graphics (SIGGRAPH Asia 2024)*, vol. 43, no. 6, p. Article 193, 2024.

[107] S. Pan, R. Zhang, Y. Liu, M. Gong, and H. Huang, "Building lod representation for 3d urban scenes," *arXiv preprint arXiv:2505.15190*, 2025.

[108] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller, J. Penna, and R. Rombach, "Sdxl: Improving latent diffusion models for high-resolution image synthesis," *arXiv preprint arXiv:2307.01952*, 2023.

[109] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, "Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 3, pp. 1623–1637, 2020.

[110] A. F. Almaz, E. A. E. El-Agouz, M. T. Abdelfatah, and I. R. Mohamed, "The future role of artificial intelligence (ai) design's integration into architectural and interior design education to improve efficiency, sustainability, and creativity," *Sustainability and Creativity*, vol. 3, no. 12, pp. 1749–1772, 2024.

[111] R. Shi, H. Chen, Z. Zhang, M. Liu, C. Xu, X. Wei, L. Chen, C. Zeng, and H. Su, "Zero123++: A single image to consistent multi-view diffusion base model," *arXiv preprint arXiv:2310.15110*, 2023.

[112] Z. Li, Y. Chen, L. Zhao, and P. Liu, "Controllable text-to-3d generation via surface-aligned gaussian splatting," in *arXiv*, 2024, arXiv:2403.09981.

[113] Y. Yang, Y. Huang, X. Wu, Y.-C. Guo, S.-H. Zhang, H. Zhao, T. He, and X. Liu, "Dreamcomposer: Controllable 3d object generation via multi-view conditions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 8111–8120.

[114] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, "Vision transformers for dense prediction," *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, miDaS monocular depth models; cite appropriate release used.

[115] H. Hu, Z. Zhou, V. Jampani, and S. Tulsiani, "Mvd-fusion: Single-view 3d via depth-consistent multi-view generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 9698–9707.

[116] M. Liu, C. Xu, H. Jin, L. Chen, T. M. Varma, Z. Xu, and H. Su, "One-2-3-45: Any single image to 3D mesh in 45 seconds without per-shape optimization," in *Advances in Neural Information Processing Systems*, 2024.

[117] Y. Liu, C. Lin, Z. Zeng, X. Long, L. Liu, T. Komura, and W. Wang, "Syncdreamer: Generating multiview-consistent images from a single-view image," *arXiv preprint arXiv:2309.03453*, 2023.

[118] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.

[119] D. Kim, "Adaptive reuse of industrial buildings for sustainability: analysis of sustainability and social values of industrial facades," Ph.D. dissertation, University of Texas at Austin, 2018.

[120] Y. Li, L. Zhao, J. Huang, and A. Law, "Research frameworks, methodologies, and assessment methods concerning the adaptive reuse of architectural heritage: A review," *Built Heritage*, vol. 5, no. 1, p. 6, 2021.

[121] QwenLM Team, "Qwen3-vl: Large vision-language models," https://github.com/QwenLM/Qwen3-VL, 2025.

[122] United Nations, Department of Economic and Social Affairs, Population Division, "World Urbanization Prospects: The 2018 Revision," United Nations, Tech. Rep., 2018.

[123] International Energy Agency, "Empowering Urban Energy Transitions," IEA, Tech. Rep., 2024.

[124] United Nations, "Transforming our World: The 2030 Agenda for Sustainable Development," United Nations General Assembly, Tech. Rep., 2015.

[125] P. Newman and J. Kenworthy, *Sustainability and Cities: Overcoming Automobile Dependence*. Washington, DC: Island Press, 1999.

[126] M. Batty, *The New Science of Cities*. Cambridge, MA: The MIT Press, 2013.

[127] L. D. Hopkins, *Urban Development: The Logic of Making Plans*. Washington, DC: Island Press, 2001.

[128] X. Pan, Z. Liu, C. He, and Q. Huang, "Modeling urban expansion by integrating a convolutional neural network and a recurrent neural network," *International Journal of Applied Earth Observation and Geoinformation*, vol. 112, p. 102977, 2022.

[129] X. Huang, A. Mallya, T.-C. Wang, and M.-Y. Liu, "Multimodal conditional image synthesis with product-of-experts gans," in *Computer Vision – ECCV 2022*, ser. Lecture Notes in Computer Science, vol. 13661. Cham: Springer Nature Switzerland, 2022, pp. 91–109.

[130] L. He and D. Aliaga, "Coho: Context-sensitive city-scale hierarchical urban layout generation," in *European Conference on Computer Vision*. Springer, 2024, pp. 1–18.

[131] Z. Zhou, J. Ding, Y. Liu, D. Jin, and Y. Li, "Towards generative modeling of urban flow through knowledge-enhanced denoising diffusion," in *Proceedings of the 31st ACM International Conference on Advances in Geographic Information Systems*, ser. SIGSPATIAL '23. New York, NY, USA: Association for Computing Machinery, 2023.

[132] Y. Liu, Y. Luo, Q. Deng, and X. Zhou, "Exploration of campus layout based on generative adversarial network," in *Proceedings of the 2020 DigitalFUTURES*, ser. CDRF 2020, P. F. Yuan, J. Yao, C. Yan, X. Wang, and N. Leach, Eds. Singapore: Springer, 2021, pp. 169–178.

[133] P. Sun, F. Yan, Q. He, and H. Liu, "The development of an experimental framework to explore the generative design preference of a machine learning-assisted residential site plan layout," *Land*, vol. 12, no. 9, p. 1776, 2023.

[134] X. Ye, J. Du, and Y. Ye, "Masterplangan: Facilitating the smart rendering of urban master plans via generative adversarial networks," *Environment and Planning B: Urban Analytics and City Science*, vol. 49, no. 3, pp. 794–814, 2022.

[135] S. Zhou, W. Jia, H. Diao, X. Geng, Y. Wu, M. Wang, Y. Wang, H. Xu, Y. Lu, and Z. Wu, "A cyclegan-pix2pix framework for multi-objective 3d urban morphology optimization: enhancing thermal performance in high-density areas," *Sustainable Cities and Society*, vol. 126, p. 106400, 2025.

[136] S. Hong, X. Zhang, T. Du, S. Cheng, X. Wang, and J. Yin, "Cons2plan: Vector floorplan generation from various conditions via a learning framework based on conditional diffusion models," in *Proceedings of the 32nd ACM International Conference on Multimedia*, ser. MM '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 3248–3256.

[137] S. Gupta, D. Samaras, and C. Chen, "Topodiffusionnet: A topology-aware diffusion model," *International Conference on Learning Representations*, 2025.

[138] L. He and D. Aliaga, "Globalmapper: Arbitrary-shaped urban layout generation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 10 2023, pp. 454–464.

[139] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[140] N. Nauata, S. Hosseini, K.-H. Chang, H. Chu, C.-Y. Cheng, and Y. Furukawa, "House-gan++: Generative adversarial layout refinement network towards intelligent computational agent for professional architects," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 13 627–13 636.

[141] I. Karadag, O. Z. Güzelci, and S. Alaçam, "Edu-ai: A twofold machine learning model to support classroom layout generation," *Construction Innovation*, vol. 23, no. 4, pp. 898–914, 2023.

[142] L. Yang, L. Li, Q. Chen, J. Zhang, T. Feng, and W. Zhang, "Street Layout Design via Conditional Adversarial Learning," *arXiv preprint arXiv:2305.08186*, 2023.

[143] Y. Liu, Z. Zhang, and Q. Deng, "Exploration on diversity generation of campus layout based on gan," in *Hybrid Intelligence*, ser. Computational Design and Robotic Fabrication (CDRF 2022), P. F. Yuan, H. Chai, C. Yan, K. Li, and T. Sun, Eds. Singapore: Springer, 2023, pp. 233–243.

[144] P. Zeng, W. Gao, J. Yin, P. Xu, and S. Lu, "Residential floor plans: Multi-conditional automatic generation using diffusion models," *Automation in Construction*, vol. 162, p. 105374, 2024.

[145] Z. Cai, M. Demuzere, Y. Tang, and Y. Wan, "The characteristic and transformation of 3d urban morphology in three chinese megacities," *Cities*, vol. 131, p. 103988, 2022.

[146] D. Patil and R. Gupta, "Spatiotemporal analysis and prediction of urban evolution patterns using ann tool," *Proceedings of the Institution of Civil Engineers – Urban Design and Planning*, vol. 176, no. 4, pp. 159–169, 2023.

[147] R. Liu, Y. Xu, C. Xue, Z. Xia, G. Li, X. Gou, and S. Luo, "Simulation of early warning indicators of urban expansion derived from machine learning," *Journal of Urban Planning and Development*, vol. 149, no. 1, p. 04022058, 2023.

[148] P. Tsagkis, E. Bakogiannis, and A. Nikitas, "Analysing urban growth using machine learning and open data: An artificial neural network modelled case study of five greek cities," *Sustainable Cities and Society*, vol. 89, p. 104337, 2023.

[149] C. Zhou, S. Zhang, B. Liu, T. Li, J. Shi, and H. Zhan, "Using deep learning to unravel the structural evolution of block-scale green spaces in urban renewal," *Cities*, vol. 150, p. 105030, 2024.

[150] A. Jaad and K. F. Abdelghany, "The story of five mena cities: Urban growth prediction modeling using remote sensing and video analytics," *Cities*, vol. 118, p. 103393, 2021.

[151] Y. Liu, C. Wu, J. Wu, Y. Zhang, X. Bi, M. Wang, E. Yan, C. Song, and J. Li, "Projected spatiotemporal evolution of urban form using the sleuth model with urban master plan scenarios," *Remote Sensing*, vol. 17, no. 2, 2025.

[152] K. Lei, M. Qin, B. Bai, G. Zhang, and M. Yang, "Gcn-gan: A non-linear temporal link prediction model for weighted dynamic networks," in *IEEE INFOCOM 2019-IEEE conference on computer communications*. IEEE, 2019, pp. 388–396.

[153] W. Para, P. Guerrero, T. Kelly, L. J. Guibas, and P. Wonka, "Generative layout modeling using constraint graphs," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 6690–6700.

[154] K. O'shea and R. Nash, "An introduction to convolutional neural networks," *arXiv preprint arXiv:1511.08458*, 2015.

[155] T. Brooks, A. Holynski, and A. A. Efros, "Instructpix2pix: Learning to follow image editing instructions," in *CVPR*, 2023.

[156] M. Wang, Z. Xiong, J. Zhao, S. Zhou, and Q. Wang, "Pix2pix-based modelling of urban morphogenesis and its linkage to local climate zones and urban heat islands in chinese megacities," *Land*, vol. 14, no. 4, 2025.

[157] S. Gupta, A. A. Sur, A. Khurana, A. Bhakat, U. Gupta, and M. Mohsin Ali, "Smartplanai: Floorplan generation using instruct-pix2pix," in *International Conference on Data Analytics & Management*. Springer, 2024, pp. 523–538.

Publications

Journal Papers

[1] Xusheng Du, Chengyuan Li, Qingpeng Li, Yuxin Lu, Yimeng Xu, Ye Zhang, Zhen Xu, and Haoran Xie, "AI-Driven Urban Evolution Forecasting: A Unified Memory-Aware Multi-Conditional Generation Framework for Sustainable Development Planning," *Sustainable Cities and Society*. (under review)

[2] Xusheng Du, Tianyu Zhang, and Haoran Xie, "DualShape: Sketch-Based 3D Shape Design with Part Generation and Retrieval," *IEEE Access*, vol. 12, pp. 18888–18900, 2024.

[3] Zhengyang Wang, Yuxiao Ren, Hao Jin, Jieli Feng, Xusheng Du, Ye Zhang, Haoran Xie, "Controllable Generation of Building Representations: Aligning Campus Building Design Intent with Multi-Stage Retrieval-Augmented Diffusion Models," *Frontiers of Architectural Research*. (accepted)

[4] Hao Jin, Hengyuan Chang, Xiaoxuan Xie, Zhengyang Wang, Xusheng Du, Shaojun Hu, Haoran Xie, "Sketch-Guided Stylized Landscape Cinemagraph Synthesis," *Computer & Graphics*. (accepted)

[5] Tianyu Zhang, Xiaoxuan Xie, Xusheng Du, and Haoran Xie, "Sketch-Guided Scene Image Generation with Diffusion Model," *Computer & Graphics*, vol. 129, p. 104226, 2025. Also in The 9th ACM/EG Expressive Symposium.

[6] Chengyuan Li, Tianyu Zhang, Xusheng Du, Ye Zhang, and Haoran Xie, "Generative AI Models for Different Steps in Architectural Design: A Literature Review," *Frontiers of Architectural Research*, 2024.

[7] Sicheng Li, Xusheng Du, Haoran Xie, and Kazunori Miyata, "Interactive Drawing Interface for Aging Anime Face Sketches Using Transformer-Based Generative Model," *IEEE Access*, 2024. doi: 10.1109/ACCESS.2024.3466230.

International Conferences

[8] Xusheng Du, Athiwat Kongkaeo, Ye Zhang, and Haoran Xie, "Automatic LoD Sketch Extraction from Architectural Models Using Generative AI: Dataset Construction for Multi-Level Architectural Design Generation," *The Association for Computer-Aided Architectural Design Research in Asia (CAADRIA 2026)*, Hsinchu, Taiwan, Apr. 2026.

[9] Warissara Booranamaitree*, Xusheng Du*, Yushu Cai, Zhengyang Wang, Ye Zhang, and Haoran Xie, "Sketch-Based Facade Renovation with Generative AI: A Streamlined Framework for Bypassing As-Built Modeling in Industrial Adaptive Reuse," *The Association for Computer-Aided Architectural Design Research in Asia (CAADRIA 2026)*, Hsinchu, Taiwan, Apr. 2026. (*Equal contribution.)

[10] Zhengyang Wang, Nuttapong Rochanavibhata, Yuxiao Ren, Xusheng Du, Ye Zhang, and Haoran Xie, "Retrieval-Augmented Sketch-Guided 3D Building Generation: Generative Architectural Design for Japanese Detached Houses," *The Association for Computer-Aided Architectural Design Research in Asia (CAADRIA 2026)*, Hsinchu, Taiwan, Apr. 2026.

[11] Xusheng Du, James Quirk, Chia-Ming Chang, Haoran Xie, Takeo Igarashi, "CoreSearch: An Interactive and Controllable Keyword Expansion Search Interface for E-Commerce," *28th International Conference on Human-Computer Interaction (HCI International 2026)*, Canada, Jul. 2026.

[12] Xusheng Du, Ruihan Gui, Zhengyang Wang, Ye Zhang, and Haoran Xie, "Multi-View Depth Consistent Image Generation Using Generative AI Models: Application on Architectural Design of University Buildings," *The Association for Computer-Aided Architectural Design Research in Asia (CAADRIA 2025)*, Tokyo, Japan, Mar. 2025.

[13] Zhengyang Wang, Hao Jin, Xusheng Du, Yuxiao Ren, Ye Zhang, and Haoran Xie, "From Architectural Sketch to Conceptual Representation: Using Structure-Aware Diffusion Model to Generate Renderings of School Buildings," *The Association for Computer-Aided Architectural Design Research in Asia (CAADRIA 2025)*, Tokyo, Japan, Mar. 2025.

[14] Xiaoxuan Xie, Xusheng Du, Minhao Li, Xi Yang, and Haoran Xie, "DiffOBI: Diffusion-Based Image Generation of Oracle Bone Inscription

Style Characters," *ACM SIGGRAPH Asia 2024, Technical Communications*, Tokyo, Japan, Dec. 2024.

[15] Zhengyang Wang, Xusheng Du, Tsukasa Fukusato, and Haoran Xie, "Video2Comic: A Dynamic Comic Editor with Video Clips," *26th International Conference on Human-Computer Interaction (HCI International 2024)*, Washington, USA, Jul. 2024.

[16] Chia-Ming Chang, Yi He, Xusheng Du, Xi Yang, and Haoran Xie, "Dynamic Labeling: A Control System for Labeling Styles in Image Annotation Tasks," *26th International Conference on Human-Computer Interaction (HCI International 2024)*, Washington, USA, Jul. 2024.

[17] Tianyu Zhang, Xusheng Du, Chia-Ming Chang, Xi Yang, and Haoran Xie, "SGDraw: Scene Graph Drawing Interface Using Object-Oriented Representation," *25th International Conference on Human-Computer Interaction (HCI International 2023)*, Denmark, Jul. 2023.

[18] Jiahao Weng, Xusheng Du, and Haoran Xie, "DualSlide: Global-to-Local Sketching Interface for Slides Content and Layout Design," *NICOGRAPH International 2023*, Hokkaido, Japan, Jun. 2023.

[19] Xusheng Du, Yi He, Xi Yang, Chia-Ming Chang, and Haoran Xie, "Sketch-Based 3D Shape Modeling from Sparse Point Clouds," *Proceedings of the International Workshop on Advanced Image Technology (IWAIT 2022)*, Jan. 2022.