Contextual Integration through Computation: Algorithmic Approaches for Incorporating Lattice Patterns into Facade Designs

Milind Raj Bajracharya[1], Sanjaya Uprety[2*]
[1,2]Department of Architecture, Institute of Engineering, Pulchowk Campus, Tribhuvan University, Nepal
[*]Corresponding Author: suprety@ioe.edu.np

Abstract

Context and technology have significantly shaped architectural design throughout history, with contextuality playing a vital role in assigning meaning to designs. In light of computational tools and algorithms that revolutionize architectural design, enabling intricate and complex building geometries within shorter timeframes and heightened efficiency, this study aims to underscore the potential of algorithmic façade design. It focuses on contextual integration, utilizing the Newari lattice window as a foundational element for design exploration. The study employed computational modeling with parametric tools, scripting, and generative algorithms to create digital models that reflect the Newari lattice window pattern and fractal geometry. It utilized Text-Based Language (TBLs) like Python and Visual-Based Language (VBLs) like Grasshopper for visual experimentation and precision. Demonstrated through the re-design of the façade of the existing commercial building, Civil Mall, a significant building in Kathmandu Valley's urban landscape, the research showcases algorithmic prowess in seamlessly integrating traditional Newari wisdom into intricate geometrical patterns. By transcending manual methods through recursion, iteration, randomness, and embracing unforeseen possibilities, the study establishes a foundational reference, enriching Nepal's architectural discourse beyond the traditional versus modern binary. This might be helpful in fostering a distinctive modern architectural language unique to Nepal. The study holds potential for further research on the prevailing traditional architectural forms and geometry in the country.

Keywords: *Algorithmic Design, Computational Design, Contemporary Architecture, Context, Façade, Pattern*

Introduction

Architectural design has witnessed a dynamic interplay between context and technology throughout history, wherein contextuality bestows designs with profound meaning. However, a notable gap exists in effectively contextualizing traditional and contemporary architectural styles within urban landscapes. Architecture is a field that evolves constantly, embracing technological advancements and innovative design approaches (Henri H, 2003). Contemporary buildings, particularly those influenced by modern and post-modern architectural movements, often lack meaningful engagement with their cultural, historical, and environmental contexts, as highlighted by Byrne (1989). According to Silva (2019) and Kalita and Kumar (2020), this disconnect between technology and context calls for a re-evaluation of design methodologies and a renewed emphasis on integrating innovative approaches, such as computational and algorithmic methods, to bridge this gap.

In the context of Nepal, despite the rich architectural character provided by the traditional Newari architecture of the Kathmandu Valley, the current architectural practice has led to a disconnect between architectural designs and their surroundings, evoking feelings of alienation among observers and occupants. Unfortunately, contemporary architects have often overlooked or misinterpreted the context, geometry, form, and architectural elements of traditional buildings that exhibit rich cultural heritage and compositional qualities, reflecting the intricate artisanship and attention to detail integral to Newari culture, including elements like woodcarving, ornate brickwork, and decorative motifs (Korn, 1979; Tiwari, 2009). The result is the chaotic urban landscape of the Kathmandu Valley characterized by mimicked architectural styles from the West.

In this context, the development of computational tools and algorithms that have revolutionized architectural design in recent years offers immense possibilities to integrate traditional essence into contemporary building forms. Caetano et al. (2022); Gürcan et al. (2023); Kalita and Kumar (2020) assert that such revolutionary technology has shown potential for the creation of intricate and complex façade systems. However, despite architectural evolution and the integration of technological advancements, a persistent disjunction remains between technological innovation and contextual relevance. This gap underscores the need to bridge traditional and contemporary architectural principles through innovative approaches. This study aims to address the identified gap by exploring the potential of algorithmic façade design as a conduit for contextual integration. It particularly emphasizes the utilization of Newari architectural elements as foundational inspiration to create contemporary designs that harmonize with cultural, historical, and environmental contexts.

The study believes that algorithmic façade design rooted in Newari architectural elements can enhance contextual integration and alleviate the disconnect seen in modern architectural designs. Employing computational modeling techniques, including parametric tools, scripting, and generative algorithms, the study creates digital models reflecting the patterns of Newari architectural elements and fractal geometry. It utilizes both Text-Based Language (TBLs), such as Python, and Visual-Based Language (VBLs) like Grasshopper, ensuring precision and visual experimentation. The practical significance of the research lies in enriching Nepal's architectural discourse beyond the traditional versus modern binary. By establishing a reference point through algorithmic façade design, it fosters a distinctive modern architectural language unique to Nepal. Furthermore, it contributes to the architectural realm by showcasing the practical implementation of algorithms and coding in crafting culturally resonant contemporary designs. The research's scope centers around algorithmic façade design using Newari architectural elements as a muse. However, limitations may exist in fully capturing the intricate nuances of cultural, historical, and environmental contexts within algorithmic design processes.

Context and Architecture

The dynamic interplay between architectural design and context has been a cornerstone throughout history, influencing how architects either adhere to or challenge contextual elements(Çizgen, 2012). Burden (2002) defines contextual design as emphasizing the significance of context in establishing architectural meanings, encompassing factors such as site, environment, and neighborhood. The concept of context, as described by the Oxford Dictionary of English, revolves around circumstances that frame an event, statement, or idea, enabling comprehensive comprehension (Gifford, 2007). Within architecture, this translates to the physical, environmental, and socio-cultural settings in which buildings exist.

Architects in different eras have presented varying interpretations in response to contextual challenges. During the modernist era, architects like Le Corbusier adopted a scientific approach, emphasizing climatic conditions, lighting, and spatial functionality while potentially overlooking local context, vernacular, culture, and history (Capon, 1999). In contrast, Burden (2002), Gifford (2007), Mumford (1972), and Thomas and Garnham (2007) advocate for merging buildings with their environmental context, evident in harmonious alignment with the natural landscape, as seen along the Italian Mediterranean Coast. Organic Architecture, championed by F.L Wright, emerged under modernism as a design philosophy embracing characteristics, materiality, and balance inherent in the environment (Gelernter, 2001; Mumford, 1972).

In the 1950s, the concept of 'Contextuality' gained prominence, with figures like Kenneth Frampton (1993) and Robert Venturi (1977) championing tradition, history, and culture's importance in architectural design. Regionalism furthered this, emphasizing local contexts regarding history, community, society, and symbolism Abel (2000). Regionalism aimed to restore lost architectural identity and meaning against modernism's placelessness attitude. Theories like Place Theory assert that space evolves into a place when people imbue it with meaning (Çizgen, 2012; Uraz & Balamir, 2006), resonating with architectural design where context's meaning emerges through people's experiences, creating a sense of belonging and dialogue within architectural elements defining spaces.

In essence, architectural design evolves through dynamic interactions between context and contemporary elements, yielding diverse approaches to address contextual considerations. The favored contemporary approach involves skillfully amalgamating modernity with historical resonance while respectfully engaging the past, resonating with Michael Davies Sotoudeh (2013) viewpoint. Penn (2007) elaborates on contextual strategies, categorizing them into four modes: literal replication, inventive adaptation within a style, abstract reference, and intentional opposition. These strategies exemplify architects' diverse navigation of contextual influences during design. As the field progresses, a nuanced understanding of these strategies and their potential applications will enrich architectural discourse, fostering innovative solutions that harmonize with the past and resonate with the present.

Computation and Algorithms

As we delve into the realm of algorithmic approaches and coding, it becomes essential to establish a foundational understanding of computing and computational design. The term "compute" encapsulates the process of problem-solving. Evans (2011) defines computing as the study of information description, processing, and implementation. At its core, a computer takes inputs, applies functions, and generates outputs. The theoretical inception of computers can be traced back to Turing machines, which were the precursors of modern computers. Terzidis (2006) sheds light on computation, defining it as the process of calculation through mathematical or logical methods. Within this context, computational design emerges as a broad domain encompassing algorithmic design, generative design, and parametric design(Caetano et al., 2020; Gradišar et al., 2022; Knippers et al., 2021; Lee et al., 2015; Nasir & Kamal, 2023).

The term "algorithm" inherently implies finiteness, indicating a series of steps that are well-defined, countable, and bounded (Cormen et al., 2001; Goodrich & Tamassia, 2001; Terzidis, 2006). Algorithms not only quantify human thought processes but also facilitate the birth of innovative ideas. Designing an algorithm is a process that transcends the traditional pen-and-paper approach, necessitating the translation of cognitive processes into tangible dimensions. Computation and algorithms, with their vast calculations, combinatorial analyses, randomness, and recursion, have the potential to evoke innovative "thought" processes previously unexplored by the human mind. One of the key processes in this involves “coding” which plays a pivotal role in algorithmic design. Evans (2011) defines coding as the abstraction of information that computers can understand. While an algorithm is a procedural problem-solving technique, coding is the process of translating an algorithm into a computer program. Once an algorithm is designed, it must be implemented using a programming language (Downey, 2015).

The integration of technology in architectural practice has sparked extensive discussions. Scholarly research (Caetano et al., 2020; Ediz & Çağdaş, 2007; Knippers et al., 2021; Lee et al., 2023) has thoroughly explored computation's advantages in architecture. Prominent computationally driven approaches like algorithmic design, generative design, and parametric design have emerged, all grounded in deduction, induction, abstraction, generalization, and structured logic (Simon, 2019). Algorithms distill logical principles into a solution plan, recognizing repetitive patterns, universal principles, interchangeable modules, and inductive links (Cormen et al., 2001). Their power lies in inferring novel insights, expanding human intellect. This study's objective is to practically employ algorithms and algorithmic design to reveal intricate façade patterns in building development. Terzidis (2006) succinctly defines an algorithm as a finite step-by-step problem-solving process. Beyond solving problems, algorithms offer pathways to explore solutions beyond traditional methods' scope.

Connections between Algorithms and Design

Terzidis's insights reveal a fascinating symbiosis between design and algorithms. His work delves into the shared theoretical underpinnings of design and algorithms. According to Terzidis (2006), design and algorithms share parallels as both involve sets of procedures that stochastically lead to achieving goals. Design, often defined as the process of inventing physical entities that exhibit new physical order, organization, and form in response to function, remains an iterative process driven by knowledge and experience. While architectural design lacks a singular theory and is considered more of an art, Salingaros (2021) introduces architectural theories rooted in mathematics and fractal growth patterns. Algorithmic approaches can be a conduit for introducing a logical design process within this framework.

The integration of computation into the design process sparks thought-provoking questions. Capra (1997) juxtaposes the abstract, holistic nature of human thought with the quantitative nature of computation in design theories. However, Terzidis (2006) posits an alternative perspective, suggesting that algorithmic processes extend human cognition. This approach aligns architectural design with a historical trajectory characterized by continuous innovation, pushing the boundaries through ingenuity, invention, and imagination. From the Renaissance period to the realms of postmodernism and deconstructivism, architectural design perpetually seeks to transcend the known by exploring the unknown, a process rooted in the amalgamation of historical knowledge and contemporary potential.

Designing Complexity through Algorithms

One of the most significant advantages that algorithms bring over traditional manual design approaches is their capacity to create intricately complex artifacts (Terzidis, 2006). Algorithms offer solutions, organization, and exploration of problems involving heightened visual or organizational complexity. Notably intricate concepts such as randomness, infinity, limit, and infinitesimal - intricate aspects of complexity - challenge human comprehension. These concepts are not metaphysical, magical, or mysterious, but rather they rely on intellectual constructs foreign to the human mind.

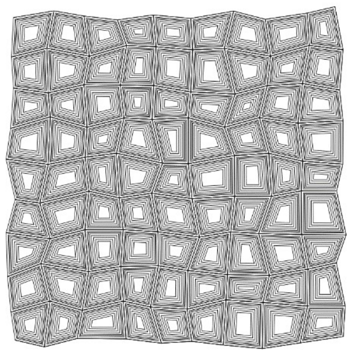

Figure 2 Recursion of a 4 Point Curve

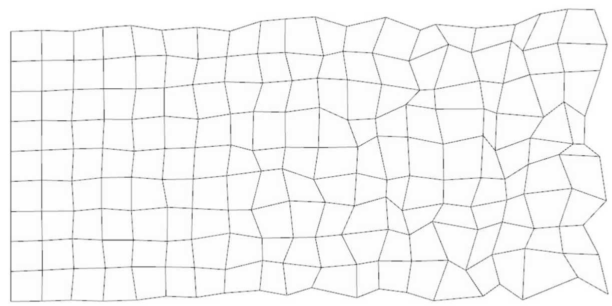

Figure 3 From Order to Chaos using Algorithms

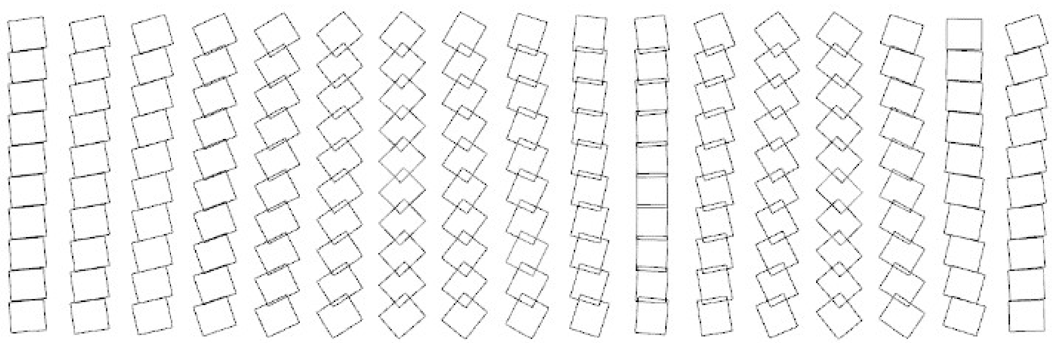

Figure 1 Transformation of a Rectangle

The theoretical foundation for crafting intricate structures through algorithms involves manipulating simple parameters across diverse facets of complexity. Recursion embodies an ontological architectural process, resulting in infinitely nested structures that self-replicate, driven by generative strategies establishing self-resemblance between the whole and its constituent parts. In the architectural realm, iteration entails the repetitive execution of instructions, leading to algorithms generating repetitive patterns that visually portray motion, change, or progress. Randomness, within architecture, denotes the absence of predictable order, pattern, or purpose, enabling algorithms to arrange elements such as windows, columns, or panels in distinct and visually captivating formations, breaking free from rigid patterns and enriching the building's overall aesthetic. The computational extension of the human mind - the computer - empowers the transcendence of quantitative complexity constraints, transforming what was once considered "impossible" in human imagination into a realm of probability.

Applications and Limits of Algorithmic Design

Algorithmic approaches find applications across a diverse spectrum, including the aforementioned complex geometry design, computational efficiency, optimization in design processes, procedural generation of innovative designs, and digital fabrication. Lee et al. (2015) serves as an illustration of algorithmic design's application. The study focuses on analyzing and modifying Islamic star patterns using digital algorithms. It introduces an efficient approach to modify and control classical geometric patterns through the utilization of computer algorithms. This study bridges the gap between classical geometric patterns and the influx of design through digital technology. It paves the way for efficiency and flexibility in future design development and material fabrication by fostering a comprehensive grasp of methods to control geometric patterns.

However, alongside its advantages, algorithmic design does encounter certain limitations. These include a steep learning curve and a dearth of expertise, especially in regions like Nepal. This approach demands expertise in both architectural design and computation methods. Additionally, challenges arise in integrating algorithmic methods with traditional design approaches, and a lack of understanding exists regarding the implications this approach carries.

Material and Methods

In this research, we have studied and analyzed the Tiki-Jhya, also known as the Lattice Window, a prominent feature in Newari architectural heritage. This window type holds a significant place within Newari communities, being a common sight on both Tiered Temples and residential buildings ("Patan Museum Architecture Galleries," 2020). The choice to focus on the Tiki-Jhya arises from its profound cultural symbolism and visual familiarity, making it an ideal candidate for design exploration that bridges traditional elements with contemporary contexts. Within domestic Newar architecture, Tiki-Jhya finds its place framing entrances with a pair of small-latticed windows, allowing minimal light and ventilation into enclosed spaces designated for storage. In contrast, the lattice windows on the first floor feature larger dimensions, aligning with concepts of privacy and protection. The lattice casements of bay windows on the second floor offer the option of being either top hung or shutter-shaped, enabling controlled openings.

Functionally, the lattice pattern in Newari windows serves as a visual barrier between interior and exterior spaces, granting occupants views outward while defining the façade's composition with its rhythmic arrangement. This pattern exhibits both cultural significance and pragmatic utility (Dangol, 2007). The study analyzed various lattice patterns found in Newari architecture, including the Square Pattern (Maka Pattern), Diamond Pattern (Ikka Pattern), and the innovative Malegwa Pattern, which ingeniously combines elements of both square and diamond patterns. Through our analysis, we aim to shed light on how these patterns have been employed historically and how they can be reimagined within contemporary design contexts using computational design and algorithms.

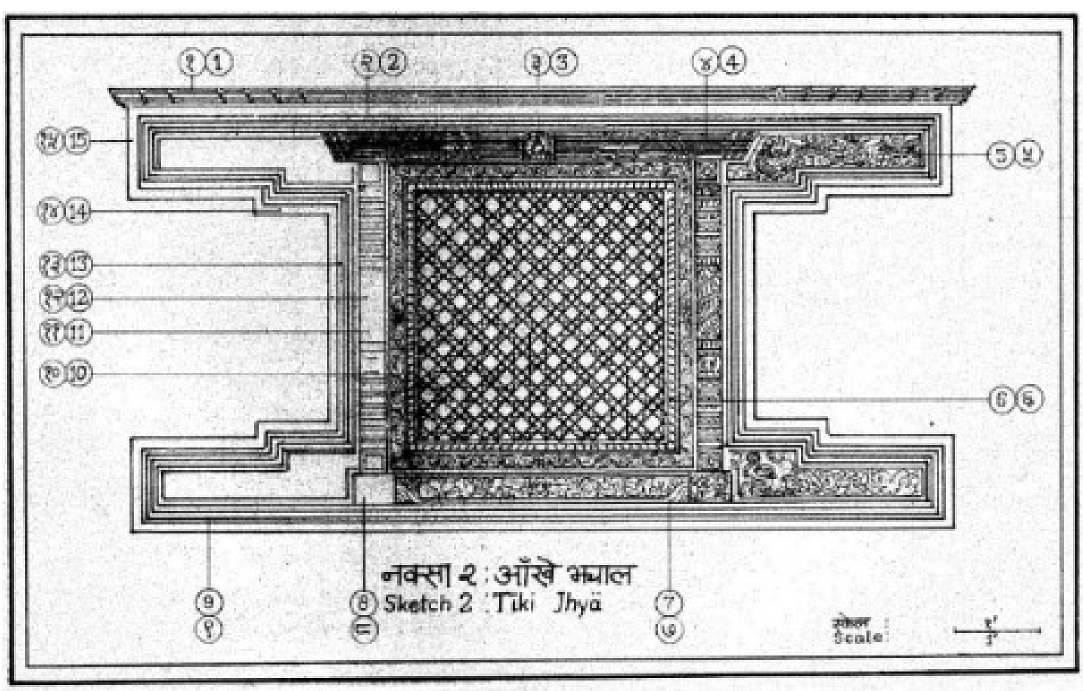

Figure 4 Illustration of Tiki Jhya by Chandra Bahadur Joshi (Joshi, 1990)

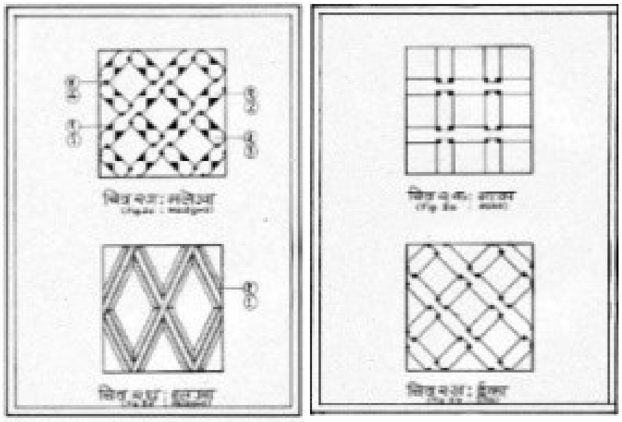

Figure 5 Illustration of Lattice Patterns by Chandra Bahadur Joshi (Joshi, 1990)

Methodology

The methodology of this paper employed a pragmatic research approach that aligned with the learning-by-doing philosophy to acquire new insights. The research-by-design strategy involved the algorithmic exploration of pattern geometry, with a specific focus on contextual design within contemporary architecture. The designer provided input for reference geometry and constraints, established the logic for pattern generation, and utilized algorithms for comprehensive exploration. The generated alternatives underwent a process of evaluation, optimization, and revision, ultimately leading to the selection of a final solution.

Computational tools and software played a pivotal role in the research, enabling the creation of digital models for lattice patterns and facade geometry. This facilitated experimentation with design parameters, shedding light on the algorithmic design possibilities for facades. Rhinoceros3D, developed by Robert McNeel & Associates, served as a versatile platform for creating curves, surfaces, and volumes using non-uniform rational B-splines (NURBS). Its compatibility with multiple file formats and mesh operations made it a foundational tool for computational modeling (Vierlinger, 2013). Both Text-Based Languages (TBLs) and Visual-Based Languages (VBLs) were utilized. TBLs, including Python, provided the means for precise and iterative looping based algorithmic design. On the other hand, VBLs, exemplified by Grasshopper, presented a visual approach, allowing architects to engage in node-based parametric modeling. Grasshopper, developed by David Rutten at Robert McNeel & Associates, facilitated script assembly through components connected by data flows. It effectively employed lists and arrays, which were crucial for algorithmic design (Vierlinger, 2013). Python scripting, in conjunction with the ghpython component in Grasshopper, revolutionized algorithmic design. Python's versatility and extensive libraries, combined with its integration into Grasshopper, empowered architects to create complex parametric models (Davidson, 2019; Rigdon-Bel, 2021). Third-party plugins also made notable contributions. LunchBox enabled the exploration of shapes, paneling, and machine learning tasks, while Parakeet focused on Algorithmic Pattern Generation, offering streamlined geometrical and natural pattern/network generation (KhalilBeigi, 2015; Miller, 2020).

The research adopted a balanced approach, integrating both text-based and visual-based methods. Text-based languages offered precision, enabling control over design and calculations. In contrast, visual-based languages provided swift real-time experimentation. This dual approach optimized design prototyping and iteration, effectively enhancing the power of algorithmic design.

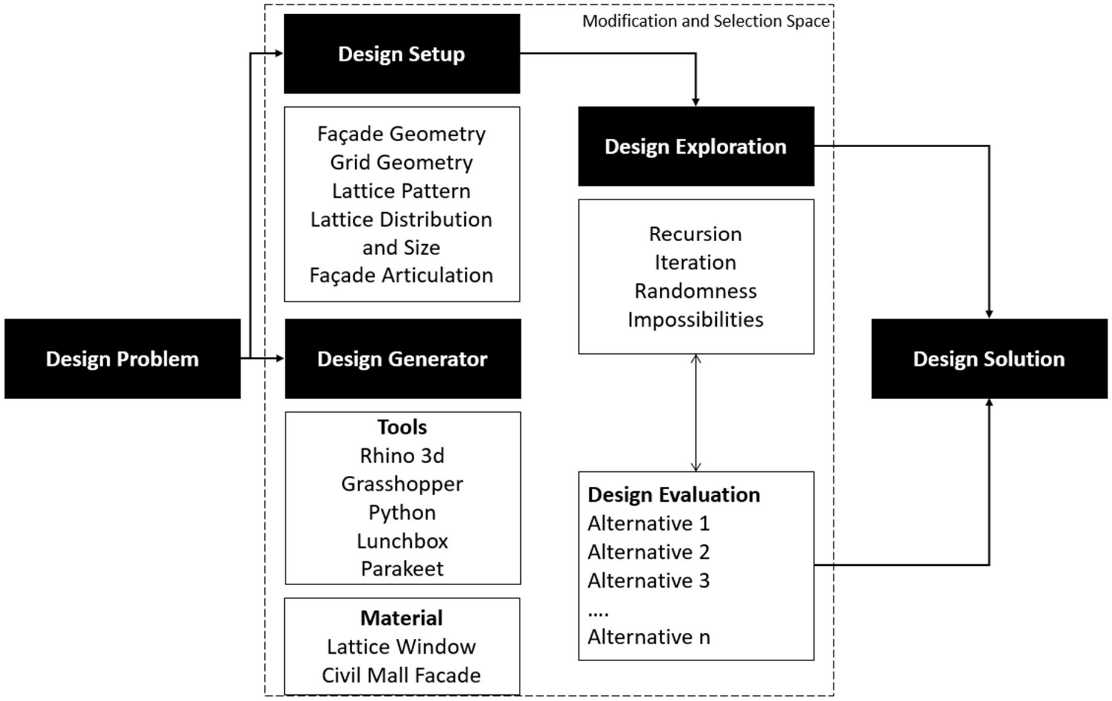

Figure 6 the Algorithmic Design Procedure of the Study

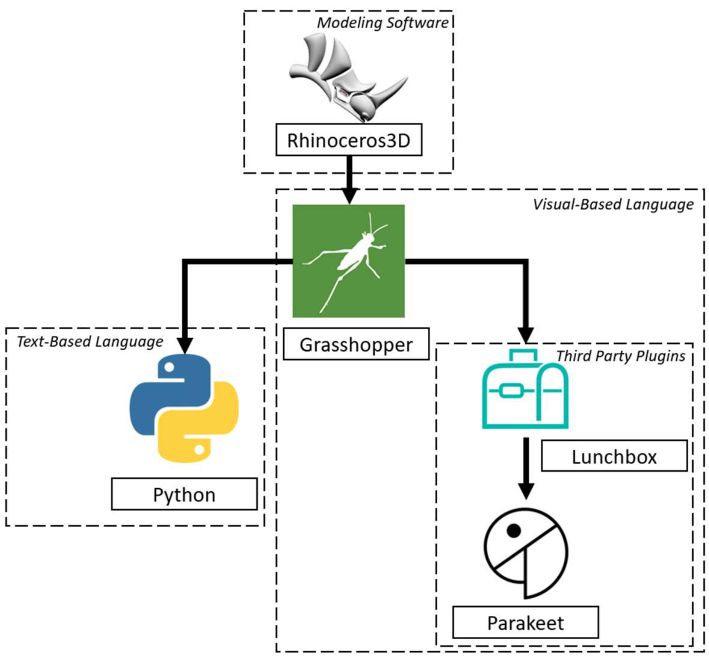

Figure 7 Algorithmic tools workflow

Demonstration Case: Redesign of Civil Mall Façade

This study centers on the Civil Mall, strategically positioned in the heart of Kathmandu and in close proximity to renowned landmarks like Dharahara, Tundikhel, and Sahid Gate (Martyr's memorial). A pioneer in the mall trend in Nepal, the mall spans 10 floors and covers a sprawling floor area of 1680 sq.m. Since its completion in 2010, it has thrived as a bustling commercial hub, attracting both the younger generation and visitors seeking shopping and leisure activities. The mall features an array of retail outlets, cafes, restaurants, a cinema hall, and a gaming zone. Its courtyard, serving as a vibrant space for public events, further solidifies its position as a significant element of the urban landscape and particularly resonates with the youth demographic.

Figure 8 Civil Mall Front Facade

The building's façade embodies an unconventional architectural style, reflecting architects' attempt to merge post-modernist elements with modern geometry and glazed windows. This fusion is further accentuated by subtle allusions to classical architecture, as manifested in its axial symmetry and neo-classical accents. However, despite its setting amidst historical landmarks like Dharahara, Tundikhel, and Sahid Gate, and the distinctive contextual architectural identity of Kathmandu Valley characterized by Newari traditional architecture, the façade design falls somewhat short of capturing these contextual influences. Its composition lacks cohesiveness, evident in the disjointed interplay of colors, materials, and volume. The dominating red hue and the lackluster unsaturated blue glaze significantly detract from the intended elegance. Moreover, materials like aluminum paneling feel incongruous, contributing to an overall sense of visual disharmony. In addition, the building's façade composition struggles to embrace a coherent design language, hinting at an endeavor to blend two architectural styles without genuinely embodying either. While some aspects echo post-modernism's geometric precision and platonic forms, the building doesn't wholly embrace either stylistic dimension. Furthermore, the absence of symbolic ties to culture or religion leaves the building disconnected from the region's rich heritage and tradition.

Considering these observations and utilizing computational design and algorithmic methodologies to develop a facade skin based on the lattice window concept, a comprehensive array of design possibilities were meticulously evaluated. The discussion of the results is aimed at identifying a design solution that not only recognizes the traditional essence but also seamlessly integrates it. To achieve this, a systematic process was adopted, as outlined in the following approach.

Design Setup

The foundational design setup was based on eight distinct dimensions, each playing a crucial role in shaping the outcome. These dimensions encompassed Facade Geometry, Pattern Geometry, Element Size and Distribution, Element Distortion, and Facade Articulation.

Facade Geometry

To establish the foundation for the facade's design, the base geometry was derived from the dimensions of Civil Mall. This involved creating a rectangle where the rectangle's height aligned with the facade's height, and its width corresponded to the facade's width. This rectangle was further divided into four sections, mirroring the building's floor divisions. This foundational geometry served as a reference point throughout the design process.

Figure 9 Transformation of Facade Geometry with incorporation of the Lattice Pattern

Grid Geometry

Within the Grasshopper environment, the lattice window pattern was crafted by referencing the underlying facade geometry. The reference geometry was subdivided into a grid layout with dimensions of 30 by 20 using a combination of division and connection components. These grids served as the foundational reference for the subsequent lattice element pattern.

Lattice Pattern

The development of the lattice pattern was bound by the constraints of the established grid geometry. The initial pattern was based on a squared lattice design (referred to as the Maka Pattern). Through an innovative approach, the square was rotated by 45 degrees from its centroid, and the edges' center point was offset within a range of 0 to 1. This transformation introduced diagonal lines at a 45-degree angle, as depicted in the figure. Notably, this geometrical manipulation led to an intriguing evolution of patterns—from the Maka Pattern to Iika Pattern, followed by the Malegwa Pattern, and eventually reverting back to the Maka Pattern.

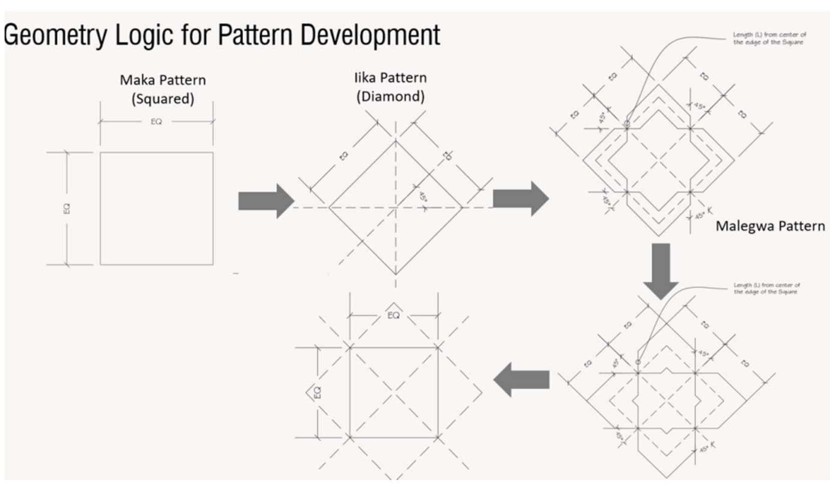

Figure 10 Geometric Logic of Lattice Pattern

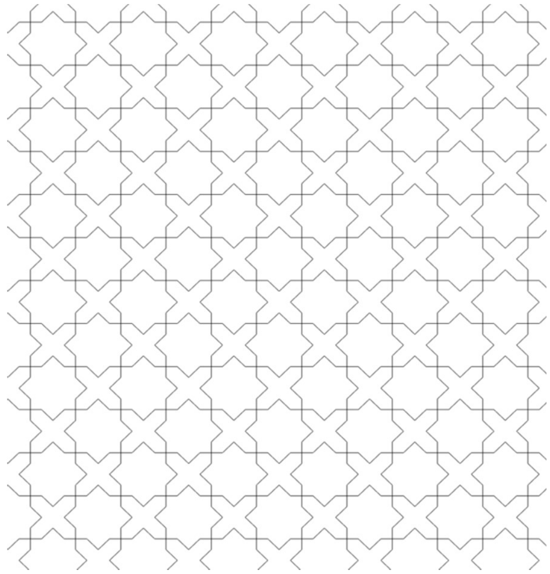

Figure 11 Algorithmic Lattice Pattern

Lattice Distribution and Size

A critical aspect of the design process involved the distribution and sizing of lattice pattern elements. The complexity of the design was directly influenced by the distribution's level. The size and replication of pattern elements were intricately linked to the chosen distribution. These adjustments adhered to the principles of complex design outlined by Terzidis (2006).

Facade Articulation

Post-generation and application of the lattice pattern, the design was further enhanced through the incorporation of additional algorithms for compositional purposes. Noteworthy additions included the introduction of a colonnade element on the ground floor. Additionally, recursive algorithms came into play to manipulate slab scaling and positioning. This intricate manipulation served to frame the design, resulting in a facade articulation reminiscent of a sloped roof, as represented in the figure.

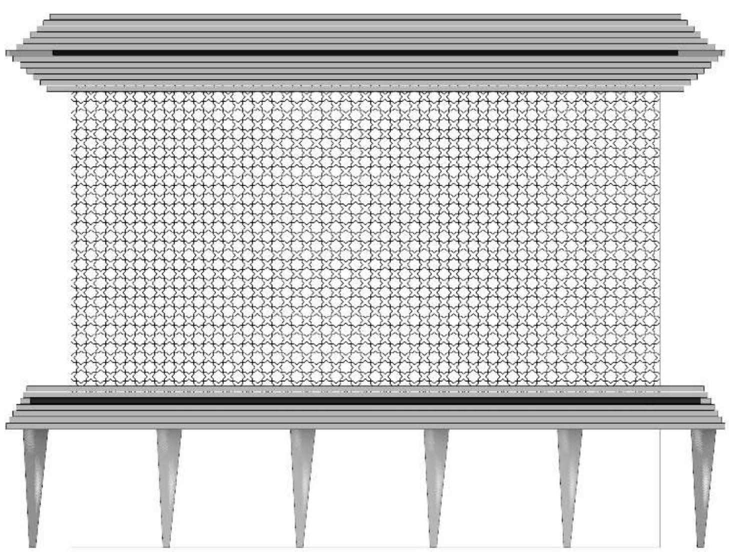

Figure 12 Facade Articulation of Lattice Pattern

Results and Discussion

The design exploration of the façade of the building embarks on an intricate integration of symbolism and abstraction derived from the lattice window pattern which according to Dangol (2007), Korn (1979) and Tiwari (2009) is intrinsic to Newari Architecture. This endeavor seeks to symbolically embed the Latticed Pattern while simultaneously rationalizing its implementation through the utilization of algorithms and coding. Central to this study is the application of an algorithmic design approach, departing from traditional methodologies in favor of computational pathways. This exploration culminates in the creation of designs marked by increased complexity and contextual sensitivity. The ensuing images represent the outcomes of the algorithmically generated façade designs. The same algorithmic process can be employed to generate akin facades, maintaining consistent proportioning and patterns, by adhering to the outlined procedures and specified materials. Through computational methods, designs that would typically require a substantial duration for conception can be produced within minutes, fostering extensive avenues for further exploration. Drawing from theoretical framework of Terzidis (2006), which elucidates various advantages of algorithmic design, the principles are extrapolated to the lattice window pattern. The ensuing results are detailed below:

Recursion

Recursive algorithms are employed to generate self-replicating patterns, as depicted in Figure 14.

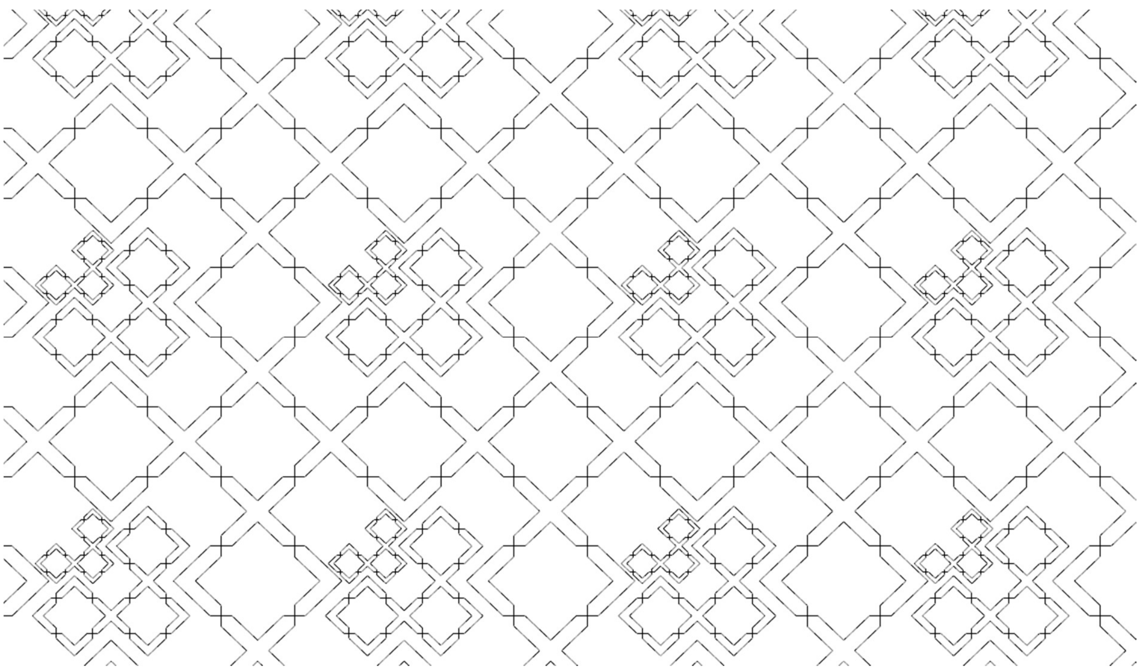

Figure 13 Utilization of Recursive Algorithm to Lattice Pattern

Iteration

The iteration principle is enacted by transforming the lattice pattern through an attractor algorithm within Grasshopper, as showcased in Figure 15. The attractor algorithms manipulate the offset length, varying it between 0 to 1 in correspondence to the distance from the attractor point. This dynamic interplay culminates in the resultant pattern.

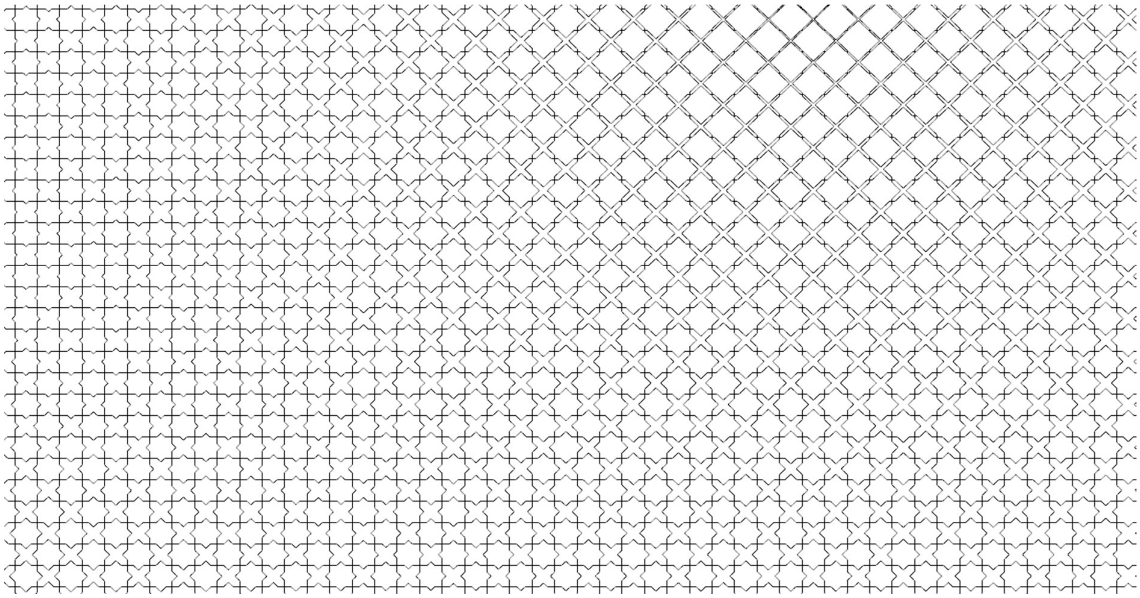

Figure 14 Iterative Transformation of Latiice Pattern

Randomness

To explore the principle of complexity, Python scripts are employed to define parameters and constraints for the random scaling of the lattice pattern, leading to the outcome illustrated in Figure 16.

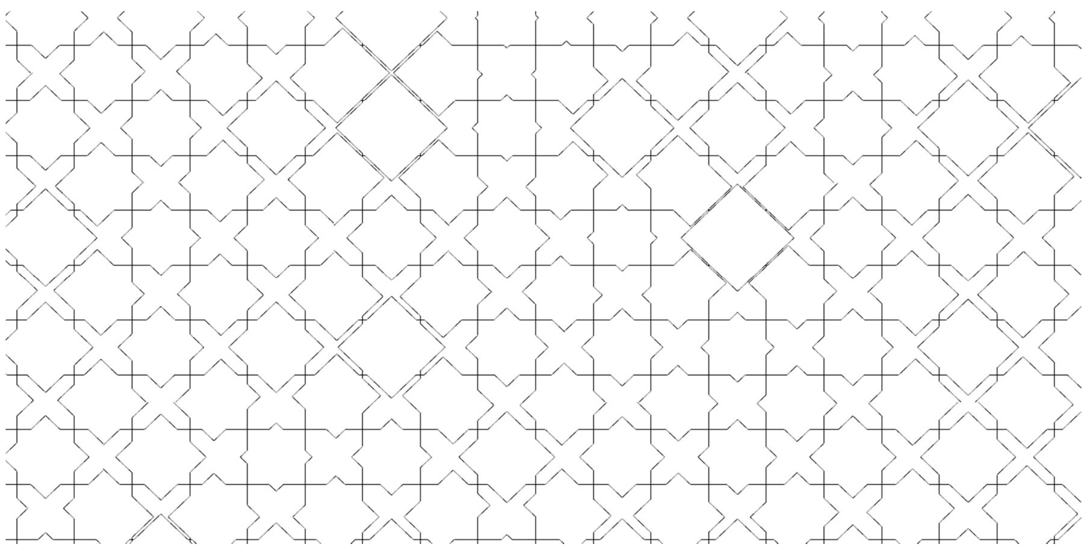

Figure 15 Randomization of Lattice Pattern

Impossibilities

This category of complexity is intriguingly paradoxical, marked by a level of unpredictability. The exploration of this principle amalgamates the complexities of recursion, iteration, and randomness. The result, depicted in Figure 17, engenders a unique outcome that transcends manual creation. The emergent patterns appear to derive from the harmonious confluence of these complexity principles. Notably, each execution of the script yields different yet captivating outcomes, allowing for iterative revisions as the designer's intent evolves.

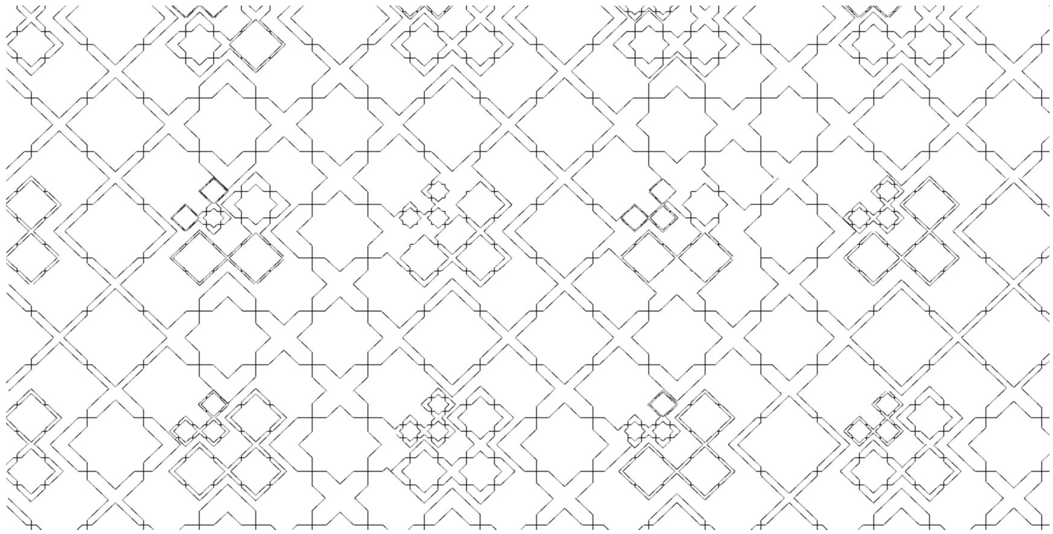

Figure 16 Combination of Recursion, Iteration and Randomness

While imperfections, such as open lines and overlapping, may surface in the outcomes, these characteristics underscore the nature of the process, potentially necessitating manual refinement by the designer. The subsequent images showcase the application of the final pattern to the façade of the Civil Mall Building, realized through 3D visualizations.

The research shows the promise of methodological approach to the exploration into algorithmic design which extends beyond traditional elements of one particular type of architectural context. This methodology creates a dialogue between traditional elements and contemporary technology for any given type of architectural context. The results parallel the work of Lee et al. (2015), who applied algorithmic approaches to Islamic motifs, and Caetano et al. (2022), who proposed a complex façade prototype. Our study underscores the efficacy of algorithmic approaches as a framework for generating patterns based on the designer's constraints, expanding creative possibilities. Nevertheless, the mastery of geometry, patterns, and algorithms rests with the designer.

While our study refrains from asserting an artistic interpretation of the lattice window pattern in the façade design, we present it as a foundational reference that highlights the extensive possibilities computational design and algorithms offer the architectural community for integrating and reinterpreting traditional architectural contexts. This research's algorithmic journey, we believe, holds the potential to ignite meaningful discussions on optimization and contextualization, equipping architects with tools to seamlessly blend tradition and innovation. Through algorithmic façade design, it effectively bridges traditional Newari elements with contemporary architecture, surpassing manual methods and nurturing a unique modern architectural language in Nepal. Beyond its immediate scope, this approach resonates globally by harmonizing tradition and technology, encouraging culturally sensitive architectural practices that honor context, culture, and heritage.

Conclusion

The research aimed to assess the viability of computational tools and algorithms for enhancing façade design with a specific emphasis on contextual integration. The study effectively demonstrated the substantial potential of computational tools and algorithms in achieving contextual integration through the showcased redesign of a commercial building's façade. While the interpretation of this integration remains open to the architectural community, the research concludes that algorithmic façade design serves as a dynamic convergence point between tradition and technology within contemporary architectural practice.

Figure 17 Architectural Visualization of Lattice Pattern over Demonstration Case Building Facade

Leveraging computational power for design generation, the research offers a notable solution to the challenge of contextual disconnection in modern architectural design. The proposed strategies go beyond simple solutions, offering diverse problem-solving methodologies that enhance efficiency and streamline the design process. Notably, these processes inherently embrace unpredictability. While computers lack the critical discernment of human designers, the algorithmic process serves as an extension of the designer's creative intellect, reflecting their unique interpretation. This approach holds tremendous promise for context-driven design exploration. Despite the complexities of harmonizing traditional and contemporary elements, this research adeptly showcases algorithmic proficiency by seamlessly integrating intricate patterns of Newari architectural heritage. As discussions progress, this study catalyzes further inquiry, particularly in façade computation and contextually rooted pattern generation. In Nepal, introducing algorithmic design to architecture introduces an innovative paradigm, with this study laying the foundation for forthcoming explorations uniting technology and context.

Conflict of Interests

The author hereby declares that there is no conflict of interest.

Reerences

Abel, C. (2000). *Architecture and identity: Responses to cultural and technological change*. Routledge.

Burden, E. (2002). *Illustrated Dictionary of Architecture*. McGraw-Hill Companies,Incorporated. https://books.google.com.np/books?id=S89PAAAAMAAJ

Byrne, E. D. (1989). DESIGN AFTER MODERNISM: BEYOND THE OBJECT. John Thackara. *Art Documentation: Journal of the Art Libraries Society of North America*, *8*(1), 34-35. https://doi.org/10.1086/adx.8.1.27948016

Caetano, I., Leitão, A., & Teixeira Bastos, F. (2022). Converting Algorithms into Tangible Solutions: A Workflow for Materializing Algorithmic Facade Designs. In (pp. 196-214). https://doi.org/10.1007/978-3-031-20241-4_14

Caetano, I., Santos, L., & Leitão, A. (2020). Computational design in architecture: Defining parametric, generative, and algorithmic design. *Frontiers of Architectural Research*, *9*(2), 287-300. https://doi.org/10.1016/j.foar.2019.12.008

Capon, D. S. (1999). *Architectural Theory, Le Corbusier's Legacy: Principles of Twentieth-century Architectural Theory Arranged by Category*. Wiley. https://books.google.com.np/books?id=eA1QAAAAMAAJ

Capra, F. (1997). *The Web of Life: A New Scientific Understanding of Living Systems*. Knopf Doubleday

Publishing Group. https://books.google.com.np/books?id=3OAIaUZQhM0C
Çizgen, G. (2012). Rethinking The Role of Context and Contextualism in Architecture and Design.
Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2001). *Introduction To Algorithms*. MIT Press. https://books.google.com.np/books?id=NLngYyWFl_YC
Dangol, P. (2007). *Elements of Nepalese Temple Architecture*. Adroit Publishers.
Davidson, S. (2019). *An Overview of the GhPython Component*. Robert McNeel & Associates. https://developer.rhino3d.com/guides/rhinopython/ghpython-component/
Downey, A. (2015). *Think Python: How to Think Like a Computer Scientist*. O'Reilly Media. https://books.google.com.np/books?id=MJ4bCwAAQBAJ
Ediz, Ö., & Çağdaş, G. (2007). A Computational Architectural Design Model Based on Fractals. *Open House International*, *32*(2), 36-45. https://doi.org/10.1108/ohi-02-2007-b0005
Evans, D. (2011). *Introduction to Computing: Explorations in Language, Logic, and Machines*. Eleven Learning. https://books.google.com.np/books?id=6i8YYAAACAAJ
Frampton, K. (1993). 20 Toward a Critical Regionalism: Six points for an architecture of resistance. *Postmodernism: a reader*, *268*.
Gelernter, M. (2001). *A History of American Architecture: Buildings in Their Cultural and Technological Context*. Manchester University Press. https://books.google.com.np/books?id=tCe8AAAAIAAJ
Gifford, R. (2007). Environmental psychology: Principles and practice.
Goodrich, M. T., & Tamassia, R. (2001). *Algorithm Design: Foundations, Analysis, and Internet Examples*. Wiley. https://books.google.com.np/books?id=BJRBETtWD8cC
Gradišar, L., Klinc, R., Turk, Ž., & Dolenc, M. (2022). Generative Design Methodology and Framework Exploiting Designer-Algorithm Synergies. *Buildings*, *12*(12). https://doi.org/10.3390/buildings12122194
Gürcan, C., Heyik, M., Taştan, H., & Zoroğlu, F. (2023). A Computational Design Strategy for Integrated Façades. *MEGARON / Yıldız Technical University, Faculty of Architecture E-Journal*. https://doi.org/10.14744/megaron.2023.55631
Henri H, A. (2003). New Design Methods for Computer Aided Architectural Design Methodology Teaching. *International Journal of Architectural Computing*, *1*(1), 72-91. https://doi.org/10.1260/147807703322467441
Joshi, C. B. (1990). Kathmandu Upatyaka Ka Kalaatmak Jhyal Haru. *Nepal Rastra Pragya Pratisthan*.
Kalita, H., & Kumar, K. (2020). Digital Technology Integration in Different Educational Fields: Design, Architecture, Tourism, and Business Engineering. In (pp. 112-131). https://doi.org/10.4018/978-1-7998-2245-5.ch008
KhalilBeigi, E. M. a. A. (2015). *Parakeet*. https://www.food4rhino.com/en/app/parakeet
Knippers, J., Kropp, C., Menges, A., Sawodny, O., & Weiskopf, D. (2021). Integrative computational design and construction: Rethinking architecture digitally. *Civil Engineering Design*, *3*(4), 123-135. https://doi.org/10.1002/cend.202100027
Korn, W. (1979). *The Traditional Architecture of the Kathmandu Valley*. Ratna Pustak Bhandar. https://books.google.com.np/books?id=MnbIzwEACAAJ
Lee, C., Shin, S., & Issa, R. (2023). Rationalization of Free-Form Architecture Using Generative and Parametric Designs. *Buildings*, *13*, 1250. https://doi.org/10.3390/buildings13051250
Lee, J.-Y., Kim, S.-W., & Jeon, Y.-C. (2015). Study of the Control of Geometric Pattern Using Digital Algorithm (with Focus on Analysis and Application of the Islamic Star Pattern). *Advances in Materials Science and Engineering*, *2015*, 1-14. https://doi.org/10.1155/2015/950232
Miller, N. (2020). *Lunchbox*. https://apps.provingground.io/lunchbox/
Mumford, L. (1972). *Roots of Contemporary American Architecture: A Series of Thirty-seven Essays Dating from the Mid-nineteenth Century to the Present*. Dover Publications. https://books.google.com.np/books?id=xbhPAAAAMAAJ
Nasir, O., & Kamal, M. A. (2023). Exploring the Role of Parametric Architecture in Building Design: An Inclusive Approach. *Facta universitatis - series Architecture and Civil Engineering*, *21*, 95-114. https://doi.org/10.2298/FUACE230114007N
Patan Museum Architecture Galleries. (2020). In P. Museum (Ed.): Patan Museum.
Penn, W. (2007). Sense of place: design guidelines for new construction in historic districts. *A Publication of the Preservation Alliance for Greater Philadelphia*.
Rigdon-Bel, D. (2021). *What is Rhino.Python?* Robert McNeel & Associates. https://developer.rhino3d.com/guides/rhinopython/what-is-rhinopython/
Salingaros, N. A. (2021). *A Theory of Architecture*. Off The Common Books. https://books.google.com.np/books?id=2QwnEAAAQBAJ
Silva, D. (2019). *DIGITAL FABRICATION -FROM TOOL TO A WAY OF THINKING Toward redefining architectural design methodology*. https://doi.org/10.52842/conf.caadria.2019.2.463
Simon, H. A. (2019). *The Sciences of the Artificial, reissue of the third edition with a new introduction by John*

Laird. MIT press.
Sotoudeh, H. (2013). Contextual Preferences of Experts and Residents: Issue of Replication and Differentiation for New Infill Design in Urban Historical Context. *21*. https://doi.org/10.5829/idosi.wasj.2013.21.9.160
Terzidis, K. (2006). *Algorithmic Architecture*. Taylor & Francis. https://books.google.com.np/books?id=G1I7-Cc2Dy0C
Thomas, R., & Garnham, T. (2007). *The Environments of Architecture: Environmental Design in Context*. Taylor & Francis. https://books.google.com.np/books?id=Aed4AgAAQBAJ
Tiwari, S. R. (2009). *Temples of the Nepal Valley*. Himal Books. https://books.google.com.np/books?id=3S9gQwAACAAJ
Uraz, T., & Balamir, A. (2006). THEMES OF PLACE AND SPACE IN DESIGN TEACHING: A JOINT STUDIO EXPERIMENT IN AMASYA. *Middle East Technical University Journal of the Faculty of Architecture*.
Venturi, R. (1977). *Complexity and contradiction in architecture* (Vol. 1). The Museum of modern art.
Vierlinger, R. (2013). *Multi objective design interface*. na.