RESEARCH ARTICLE

Open Access

Text-to-building: experiments with AI-generated 3D geometry for building design and structure generation

Giuseppe Bono[1,2]*

Abstract

The paper seeks to investigate novel potentials for building design and structure generation that arise at the intersection of computational design and AI-generated 3D geometries. Although the use of AI technologies is exponentially increasing inside the architectural discipline, the design of spatial building configurations using AI-generated 3D geometries is still limited in its applications and represents an ongoing field of investigation in advanced architectural research. In this regard, several questions still need to be answered: how can we design new building typologies from AI-generated 3D geometries? And how can we use these typologies to shape both the real and the virtual world?

The paper proposes a new approach to architectural design where artificial intelligence is used as the starting point for design exploration, while computational design procedures are employed to convert AI-generated 3D geometries into building elements – such as columns, beams, horizontal and vertical surfaces. The paper starts with a general overview of the current use of artificial intelligence inside the architectural discipline, and then it moves towards the explanation of specific AI generative models for 3D geometry reconstruction and representation. Subsequently, the proposed working pipeline is analysed in more detail – from the creation of 3D geometries using generative AI models to the conversion of such geometries into building elements that can be further designed and optimised using computational design tools and methods. The results shown in the paper are achieved using Shap-E as the main AI model, though the proposed pipeline can be implemented with multiple AI models. The paper ends by showing some of the generated results, finally adding some considerations to the relationship between human and artificial creativity inside the architectural discipline.

The work presented in the paper suggests that the use of computational design tools and methods combined with the tectonics of the latent space opens new opportunities for topological and typological explorations. In a time where traditional architectural typologies are moving towards stagnation due to their inability to satisfy new human needs and ways of living, exploring AI-based working pipelines related to architectural design allows the definition of new design solutions for the generation of new architectural spaces. In doing so, the serendipitous aspect of AI biases is used as an auxiliary force to inform design decisions, promoting the discovery of a new inbuilt dynamism

*Correspondence:
Giuseppe Bono
gio.bono@tpbennett.com; giuseppe.bono@polimi.it
Full list of author information is available at the end of the article

between human and artificial creativity. In a time where AI is everywhere, understanding the measure of such dynamism represents a key aspect for the future of the architectural discipline.

Keywords Artificial intelligence, Computational design, Architectural design, Mesh generation, Voxelization, Structure generation

1 Introduction

We train our AI models; thereafter they train us. This could be the sentence of a modern computer scientist who, paraphrasing the famous Churchill's words, tries to synthesise what it means to be living and working in the current age of artificial intelligence (AI); a time where AI is everywhere – on our phones, our computers, our homes, our cars – and the primacy of AI-based technologies has the potential to completely revolutionise our approach on multiple aspects of reality such as security, economics, order, and even knowledge itself (Kissinger et al., 2021).

Although, when someone mentions AI these days, chances are that they are referring to learning systems – in particular, deep learning – artificial intelligence is currently used in almost every single field of development related to the current digital society, up to the point that someone – like Ray Kurzweil – has already predicted that the exponential development of AI will lead to an explosion of intelligence (Kurzweil, 2005). In technical terms, this will mean allowing AI researchers to pursue the long-term goal of creating machines that exhibit human-like intelligence – the so-called Artificial General Intelligence (AGI). In fact, although nowadays the popularity of AI is constantly increasing, some have already argued that there is no "I" in the current AI, particularly if we insist on comparing artificial intelligence to human intelligence. As explained by Jeff Hawkins, there are numerous ways that today's artificial intelligence falls short compared to human intelligence. For instance, humans learn continuously while in contrast, deep learning networks can certainly outperform humans on specific tasks, but to do that, they must be fully trained before they can be deployed; and once they are deployed, they can't learn new things on the go – at least, not yet. In other words, Hawkins states that current machines, to be considered truly intelligent, must be "machines that can rapidly learn new tasks, see analogies between different tasks, and flexibly solve new problems" (Hawkins 2021, 120). Achieving such a target certainly requires an enormous level of computational resources, something that perhaps can pause the development of AI systems once again, leading research into a new AI winter.

Speculating on the possibility of new AI summers or winters is certainly not the purpose of this paper. Nevertheless, it is important to highlight the fact that the development of AI systems has historically been based on the alternation between AI summers and AI winters – in this regard, Stanislas Chaillou depicts an interesting timeline highlighting the different periods and their relationship with the architectural discipline (Chaillou, 2022). More specifically, in the past years the use of artificial intelligence has proceeded through two main approaches in developing intelligence systems: expert systems and learning systems. Expert systems became popular in the 1970s-80s, whilst learning systems became mainstream during the first decade of the 21st century and the advent of the deep learning revolution. Expert systems are knowledge systems composed of a knowledge-based part – made by rules – and a computational engine which uses the rules to derive new results. Learning systems are systems based on neural networks, and such networks derive solutions from raw data: in other words, while expert systems require inputs and rules to derive results, learning systems require inputs – and in certain cases results – to derive rules. Such a difference, although implicit, is quite significant, since in it lies the new potential to amplify the intelligence of new computational systems, namely moving from computational workflow based on automation to new ones based on interaction and augmentation. The idea of "intelligence amplification" is certainly not new, and it has been prefigured many times in the past; for instance, Garry Kasparov referred to this when he explained the idea "to use information technology as a tool to enhance human decisions instead of replacing them with autonomous AI Systems" (Kasparov, 2017). Nowadays, such amplification is becoming a reality, and it allows a new level of influence and interaction between human intelligence and artificial intelligence.

In such a background of significant evolution, three main lines of research characterised the use of AI inside the architectural discipline: the first one is related to simulation, the second one to optimisation, and the third one refers to design.

Regarding the use of artificial intelligence for simulation procedures, relevant applications have been developed for environmental performance calculations based on surrogate models rather than traditional simulation systems, or agent-based robotic fabrication procedures where hard-coded behavioural rules are replaced by self-learning and intelligent agent behaviours. For instance,

the City Intelligence Lab (CIL) of the Austrian Institute of Technology (AIT) has developed an innovative platform InFraRed (Intelligence Framework for Resilient Design) which combines parametric and generative design, machine learning and augmented reality to enable seamless design-decision framework with real-time environmental performance feedback (Galanos et al., 2021); regarding agent-based robotic fabrication, multiple research projects are conducted at the Institute for Computational Design and Construction (ICD) of the University of Stuttgart where robots are taught construction behaviours relative to certain construction targets – such as the use of deep reinforcement learning to teach a mobile robot the control policy for elastically bending bamboo bundles into design configuration (Lochnicki et al., 2021).

In terms of optimisation, artificial intelligence is used in multiple AEC-related applications such as optimising building mass generation during early-stage design or producing floor plan layouts based on adjacencies and target areas. In this regard, an interesting comparative study has been conducted at the Department of Architecture of Texas A&M University where different AI methodologies have been analysed and compared to offer new perspectives for automated spatial layout planning (ASLP). The systematic review of ASLP+AI methods and procedures highlights the fundamental components of AI applications according to multiple approaches: image-based, graph-based, performance-based, and agent-based; including statistical models in traditional ML, DL, and RL (Ko et al., 2023).

The third and last use of AI inside the AEC industry is related to creative applications for building design such as image recognition, design exploration, and data poetics. In this regard, it is interesting to highlight the fact that the implementation of artificial intelligence inside design methods and procedures corresponds with a new posthuman tendency in architecture, namely the intention to provide new responses to a new phase of human evolution, a time where human bodies are developing in new ways, and human minds are altered in different neurological and epistemological ways by the advent of new technologies. The vision of a new posthuman design ecology can be investigated in multiple ways; for instance, it can be based on a new ecosystem composed of multiple AI models – such as the DeepHimmelblau network which can learn a significant amount of semantic characteristics and create detailed design interpretations (Prix et al., 2022) – or by investigating the idea of plasticity inside the machine intelligence and its genetic fallibility – as shown in the research conducted by Immanuel Koh (Koh, 2022). Another interesting example of the use of AI for architectural design is represented by the work conducted by

Matias Del Campo and Sandra Manninger in their architectural office SPAN. In his book *Neural Architecture*, Del Campo presents some of the studio works to illustrate the integration of architecture and artificial intelligence, emphasising the fusion of AI with traditional humanistic practices in architecture (Del Campo, 2022).

Although significantly different, all three approaches – simulation, optimisation, and design – exemplify a new working methodology currently developing inside the architectural discipline; a new open-ended and non-linear approach based on predictions, translations, and disconnections with the more conventional computational approach where parametric models and standardised automation often confines design to be a mere selection process rather than being a truly generative procedure. The current paper intends to work at the intersection between computational design and artificial intelligence, bridging the gap between radical AI approaches with traditional computational pipelines. In doing so, the idea of pursuing a radical open-ended and non-linear design approach based on AI is combined with automated systems and processes obtained through consolidated computational simulation and optimisation procedures.

Before explaining the working pipeline proposed in this paper, the next section will examine some examples related to AI generative pipelines for 3D geometry design augmentation to show the current state of the art in relation to the subject.

2 Related Work

The first part of the proposed pipeline consists of the generation of 3D geometries using AI models. The use of artificial intelligence for generative approaches inside the three-dimensional space represents an ongoing field of research currently composed of several techniques and methodologies developed to address the three-dimensional generation problem in multiple ways: acting at the intersection between neural rendering and view synthesis, using systems of photorealistic view synthesis, focusing on 3D shape representations that accommodate learning-based 3D reconstruction procedures.

In the first instance, the neural radiance fields (NeRF) address the problem of view synthesis by directly optimising the parameters of a continuous 5D scene representation. The developed algorithm converts a set of images of an object – taken from different locations in space – into an accurate 3D representation of the object itself. The algorithm uses a fully connected deep network, which has a single continuous 5D coordinate system as input – composed by the spatial location (x, y, z) and viewing direction (θ, ϕ) – and outputs the volume density and view-dependent emitted radiance at that spatial location. The final 3D geometry is the result

of a synthetic process where views are synthesised by querying 5D coordinates along camera rays and using classic volume rendering techniques to project the output colours and densities into an image (Mildenhall et al., 2020).

Always related to neural radiance fields, multiple types of frameworks have been developed to generate 3D geometry starting from 2D images. Two of them are represented by pixelNeRF and SinNeRF. In the first case, the prediction of neural radiance fields starts from one or a few images proceeding in a feed-forward manner. The input image is first encoded into a pixel-aligned feature grid; then, points are rendered along a target array and for each 3D point the feature grid is queried at the projected pixel coordinate in the input image and passes the image feature along with the coordinates of the point in view space into the network to get colour and opacity (Yu et al. 2021); in the second case, a novel semi-supervised framework can train neural radiance fields given only a single reference image as input (Xu et al., 2022).

In addition to neural radiance fields, other methods deal with 3D geometry generation using AI technology. One example always related to radiance fields but without the use of neural networks is represented by Plenoxels, a system for photorealistic view synthesis. Starting from a set of images, a sparse voxel grid is reconstructed with density and spherical harmonic coefficients at each voxel. The visual ray is rendered by computing the colour and opacity of each sample point via trilinear interpolation of the neighbouring coefficient. The voxel coefficients can then be optimised with standard MSE reconstruction loss relative to the training images (Yu et al. 2021). Focusing on the problem of learning-based 3D reconstruction, another method called DefTet (Deformable Tetrahedral Meshes) intends to overcome certain limitations constituted by implicit function representations of point cloud, voxel, and surface mesh using volumetric tetrahedral meshes for the reconstruction problem. In doing so, given an input image or point cloud, DefTet utilises a neural network to deform the vertices of an initial tetrahedron mesh and to predict the occupancy for each tetrahedron based on the input data (Gao et al., 2020). Another interesting example is represented by Get3D, a generative model that directly creates textured 3D meshes with complex topology, rich geometric details, and high-fidelity textures. In this case, a 3D SDF (Signed Distance Field) and a texture field are generated via two latent codes. 3D surface meshes are then extracted from the SDF utilising DMTet – Deep Marching Tetrahedra (Shen et al., 2021) – and query the texture field at surface points to get colours. A rasterization-based differentiable renderer is used to obtain RGB images and silhouettes. At the end of the process, two 2D discriminators are utilised to classify whether the inputs are real or fake (Gao et al., 2022).

Finally, other methods and algorithms allow the generation of 3D geometries through a more direct Text-To-3D approach. One of these methodologies is represented by CLIP-Mesh, a technique to generate a 3D model using only a target text prompt. The method consists of the deformation of the control shape of a limit subdivided surface along with its texture map and normal map to obtain a 3D asset that corresponds to the input text prompt, and it relies only on a custom pre-trained model that compares the input text prompt with multiple rendered images of the 3D model (Khalid et al., 2022). Always following a Text-To-3D approach to use AI for 3D geometry generation, OpenAI has developed two systems for 3D asset generation: Point-E and Shap-E. In the first case, the initial text prompt is fed into a GLIDE model to produce a synthetic rendered view; the rendered view is then passed to a point cloud diffusion stack which conditions the image to produce a 3D RGB point cloud geometry (Nichol et al., 2022). In the second case, Shap-E is a conditional generative model that directly generates the parameters of implicit functions that can be rendered as both textured meshes and neural radiance fields (Jun et al., 2023). For the purpose of this paper, the results are obtained by using Shap-E as the main AI model to generate the initial 3D geometries; therefore, further analysis of the technical principles is due to explain their applicability and the reason why the model has been chosen.

Shap-E is a conditional generative model designed for 3D asset creation, combining advanced neural network methodologies to offer both efficiency and versatility in generating 3D geometries. The process is divided into two main parts: firstly, an encoder is trained to map 3D assets into the parameters of an implicit function; secondly, a conditional diffusion model is trained on the outputs of the encoder. Here are the technical principles of Shap-E in the academic context:

– *Implicit Neural Representations (INRs)*: Shap-E leverages INRs to represent 3D assets. These representations map 3D coordinates to specific information like colour and density, making them resolution-independent and suitable for various applications like style transfer or shape editing.
– *Neural Radiance Fields (NeRF) and Signed Distance Functions (STF)*: Shap-E uses NeRF for mapping coordinates to densities and RGB colours, facilitating the rendering of 3D scenes from arbitrary views. It also uses STF to represent textured 3D meshes, mapping coordinates to colours and distances.
– *Encoder and Diffusion Model Training*: Shap-E employs a two-stage training process. Initially, an

encoder deterministically maps 3D assets to implicit function parameters. Then, a conditional diffusion model is trained on these outputs. This two-fold approach enhances Shap-E's capacity to generate complex and diverse 3D assets efficiently.

– *Scalability and Flexibility*: Compared to explicit 3D models, Shap-E's architecture allows for a more scalable approach to 3D generation. It converges faster and can handle a broader range of output representations. This scalability is crucial for handling large datasets and diverse asset types.
– *Latent Diffusion and Conditional Generation*: Shap-E uses latent diffusion models to generate continuous latent spaces. This method allows for the efficient generation of high-resolution 3D assets, conditioned on external data like images or text descriptions.

In terms of applicability, several points can be highlighted about the use of Shap-E:

– *Diverse 3D Asset Generation*: Shap-E is capable of generating a wide variety of 3D objects and scenes, making it suitable for applications in virtual reality, gaming, and 3D animation.
– *Efficiency in Rendering*: The model's ability to represent 3D assets as NeRFs and meshes allows for efficient rendering processes, which is beneficial in real-time applications.
– *Text-to-3D Generation*: Shap-E can generate 3D assets from textual descriptions, making it an innovative tool for designers and artists who can articulate ideas in text and witness their real-time 3D representation.
– *Open-Source Resource*: Shap-E can be freely accessed and downloaded. This aspect facilitates its use and integration with existing personal pipelines.
– *Training on Large Datasets*: Shap-E's efficiency in handling large datasets makes it a valuable tool in scenarios where vast amounts of 3D data are processed, such as in architectural modelling and urban planning.

Although a certain level of fallibility and inconsistency of the provided 3D outcomes still represents an important feature in the current state of Shap-E – a feature which certainly required further research and development – Shap-E represents a valid choice for the purpose of this paper, namely presenting a pipeline to convert AI-generated 3D geometries into building elements. Nevertheless, it is important to highlight that the explained methodology is not related to Shap-E and can be used with other AI models as well. To control the length of the paper and focus the attention on the overall pipeline, the following results related to AI-generated 3D geometries are limited to the outcomes obtained by using Shap-E while results obtained by using other AI models have not been included.

3 Methodology

The proposed methodology is based on the working pipeline explained in Figs. 1 and 2. The pipeline is composed of 5 steps:

1. *Mesh Generation*
2. *Mesh Analysis*
3. *Voxel Generation*
4. *Voxel Optimisation*
5. *Structure Generation*

The adopted working platforms and coding languages are Google Colab and Python for the first two steps (Mesh Generation and Operations), and Rhinoceros+Grasshopper and C# for the following three steps (Voxel Conversion, Voxel Optimisation, and Structure Generation).

Figure 2 shows the working pipelines in more detail. The pipeline is divided into two main parts: the first one is related to AI-generated mesh geometries, and the second one is related to the conversion of the chosen mesh into building elements. In the first part, a set of latents – namely representations of the target 3D geometry at different locations inside the latent space – are generated, interpolated, and converted into mesh geometries. Such meshes are then visualised and analysed before being exported. Once exported, the meshes are converted into voxelized geometries which are firstly optimised and then converted into structural building elements by isolating selected topological components (vertices, edges, and faces). Finally, the building elements are analysed and the final building configuration is obtained.

The following paragraphs will explain such details and will show the obtained results. It is important to highlight that the results illustrated in this paper are indicative only and based on the AI model adopted. Multiple configurations and design results can be achieved with the proposed methodology according to the chosen AI model and the topological elements that are considered to generate building assemblies.

3.1 STEP 1 – AI-Generated 3D Geometry

The results shown in this paper have been obtained by using Shap-E as the AI model to generate 3D geometries starting from a text prompt (Jun et al., 2023). The initial step is to generate latents starting from a single prompt. Figure 3 shows the results generated from the manipulation of three main inputs: batch
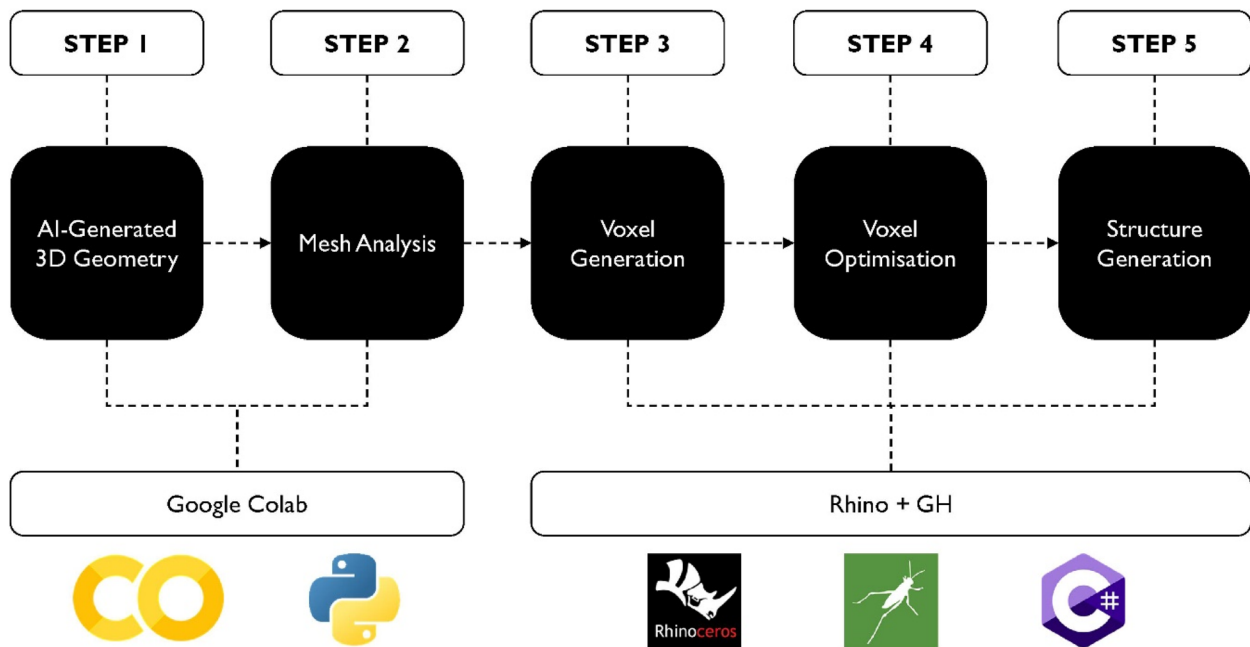
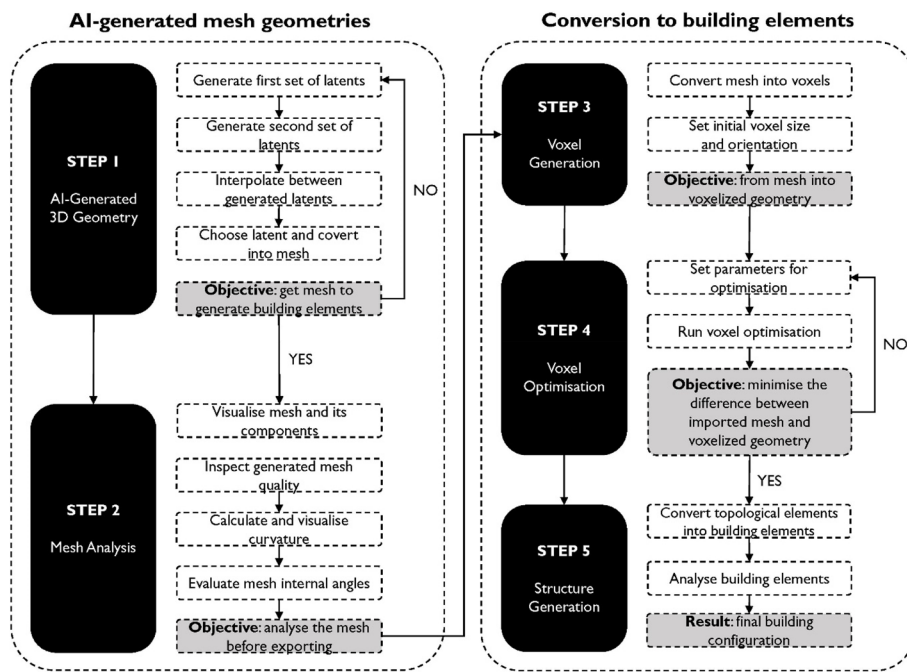**Fig. 1** Proposed working pipeline and platforms

Fig. 2 Details of the proposed working pipeline

size (= number of latents), guidance scale (= weight to text prompt), and text prompt. In this case, two sets of latents have been generated by using two different prompts and considering five options for each prompt.

Once the first set of latents is generated, a second prompt is introduced to generate a second set of latents to be used to create a linear interpolation between different latents – one for the first prompt and one for the second
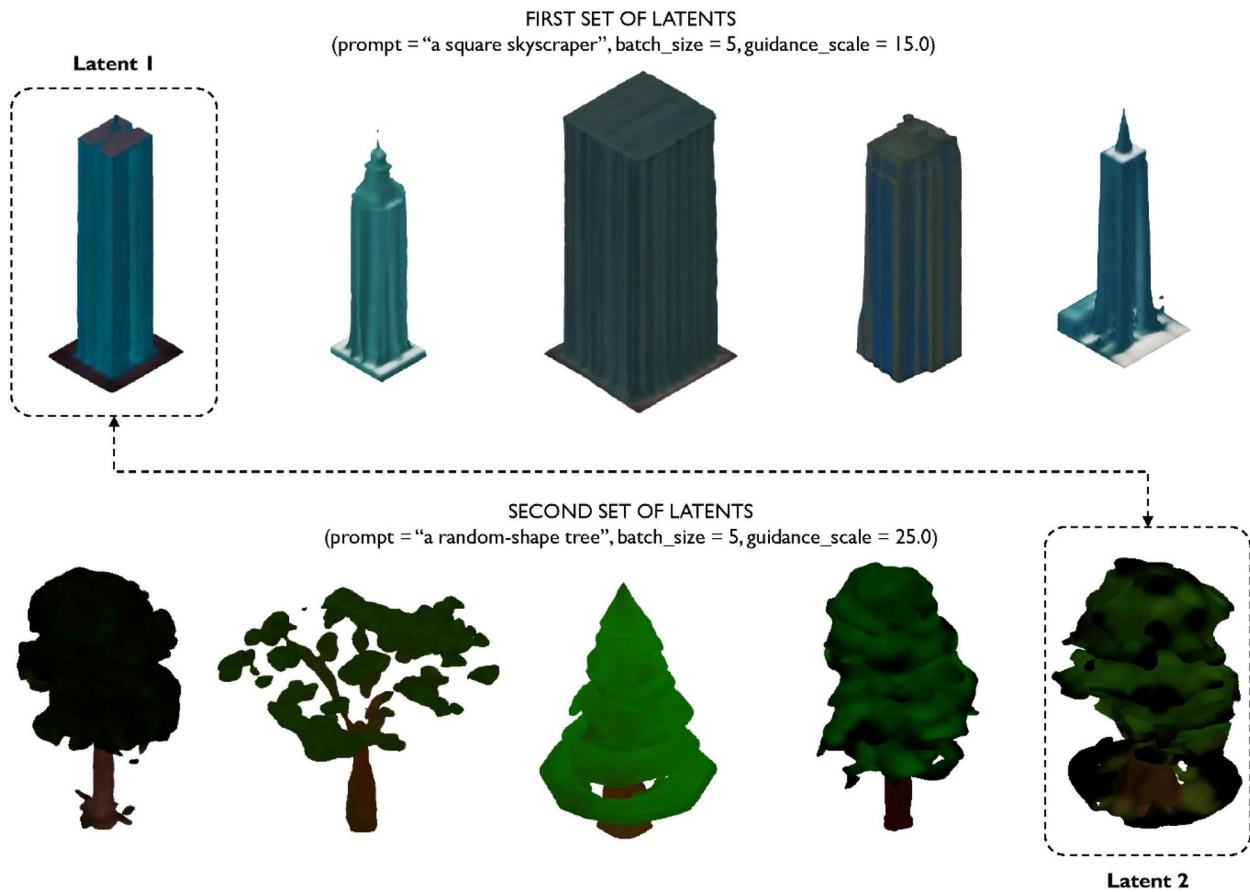
FIRST SET OF LATENTS
(prompt = "a square skyscraper", batch_size = 5, guidance_scale = 15.0)

Fig. 3 Latents generated with the first prompt and the second prompt. The dashed boxes highlight the two latents that have been chosen to perform the linear interpolation
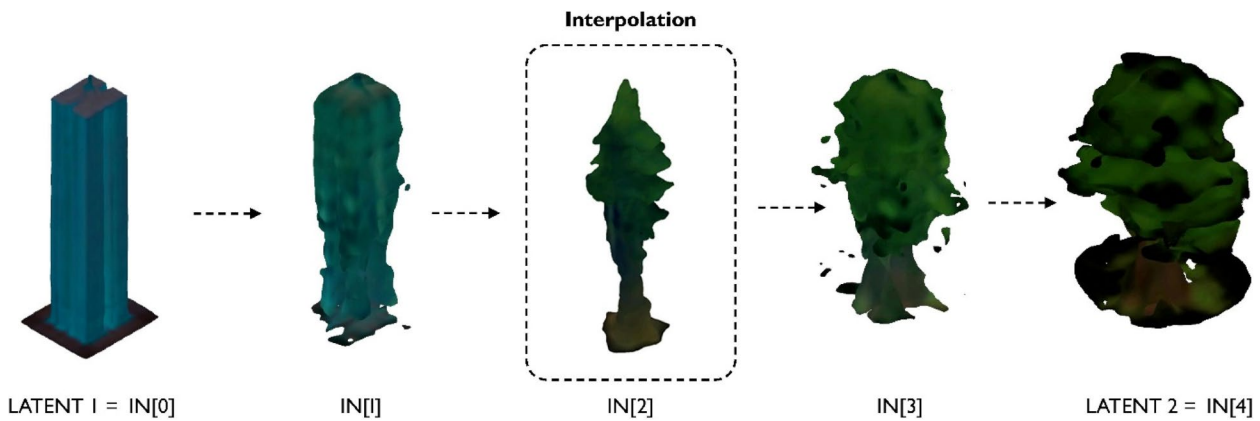
SECOND SET OF LATENTS
(prompt = "a random-shape tree", batch_size = 5, guidance_scale = 25.0)

Interpolation

LATENT 1 = IN[0]   IN[1]   IN[2]   IN[3]   LATENT 2 = IN[4]

Fig. 4 Interpolation between the two structured latents. The dashed box highlights the interpolation that has been chosen as the final mesh to analyse and import into Rhino for the following voxelization procedures

prompt (Fig. 4). This paper only shows the interpolation between two latents, although the same methodology can be used with multiple prompts and the interpolation can be performed on multiple sets of latents.

3.2 STEP 2 - Mesh analysis

To obtain the final geometry to use for visualisation, evaluation, and statistics operations, the two structured latents are interpolated and a series of options are

generated. Figure 4 shows the results of the linear interpolation divided into five steps. The number of steps is set by the user, and they can be as many as needed. The interpolation is performed to explore the tectonics of the latent space and obtain new and emergent spatial configurations to use in the following steps of the proposed pipeline – namely voxelization and building elements generation.

Once the interpolation procedure is completed, one option is chosen from the set and converted into a mesh to perform geometrical operations based on visualisation and analysis. Figure 5 shows four diagrams generated to visualise and analyse the chosen mesh and the statistics related to it. Multiple types of analysis can be generated with the proposed pipeline.

Once the analysis is completed, the final mesh is then downloaded and imported into Rhinoceros to be converted into building elements using custom C# scripts developed inside the Grasshopper working environment. The following section will show the results obtained by the adoption and development of custom computational methods and procedures.

4 Results

In the previous section, we have analysed the first two steps of the proposed working pipeline. In this section, we show the results of the conversion of the AI-generated 3D geometry into building elements.

4.1 STEP 3 - Voxel generation

To convert the AI-generated geometry into building elements, the input mesh has been converted into voxels by using a custom C# algorithm able to divide the input mesh into voxels based on set dimensions and rotation values within the x, y, z coordinate system (Fig. 6).

The developed C# algorithm can be explained with the following pseudocode:

Function 1: *RemovePartsOfMesh*

Inputs: baseMesh, percentageThreshold

Output: reducedMesh

Begin

Initialise an empty Mesh reducedMesh

Get the count of components and separate parts of baseMesh

For each part in the separated parts of baseMesh

    If the count of components in part > percentageThreshold of total components

        Add part to reducedMesh
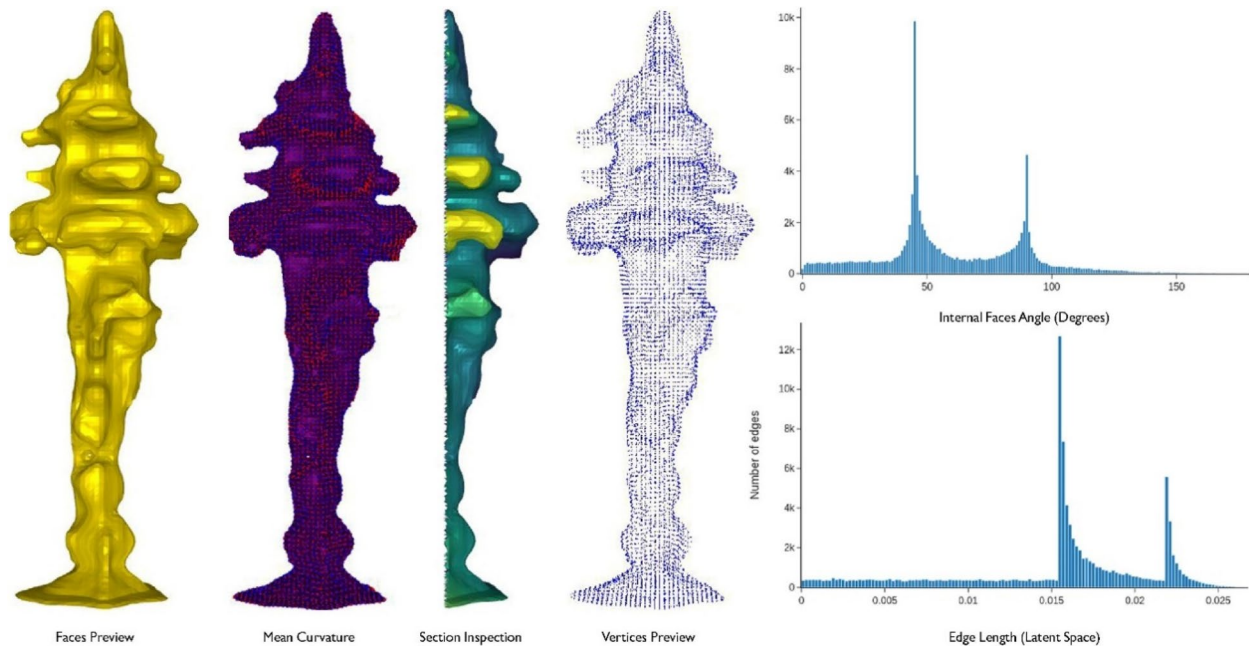
    Return reducedMesh

End

Fig. 5 Diagrams for visualisation and analysis of the choose mesh (1.Faces Preview; 2.Mean Curvature; 3.Section Inspection; 4. Vertices Preview; 5.Internal Faces Angle; 6.Edge Length)
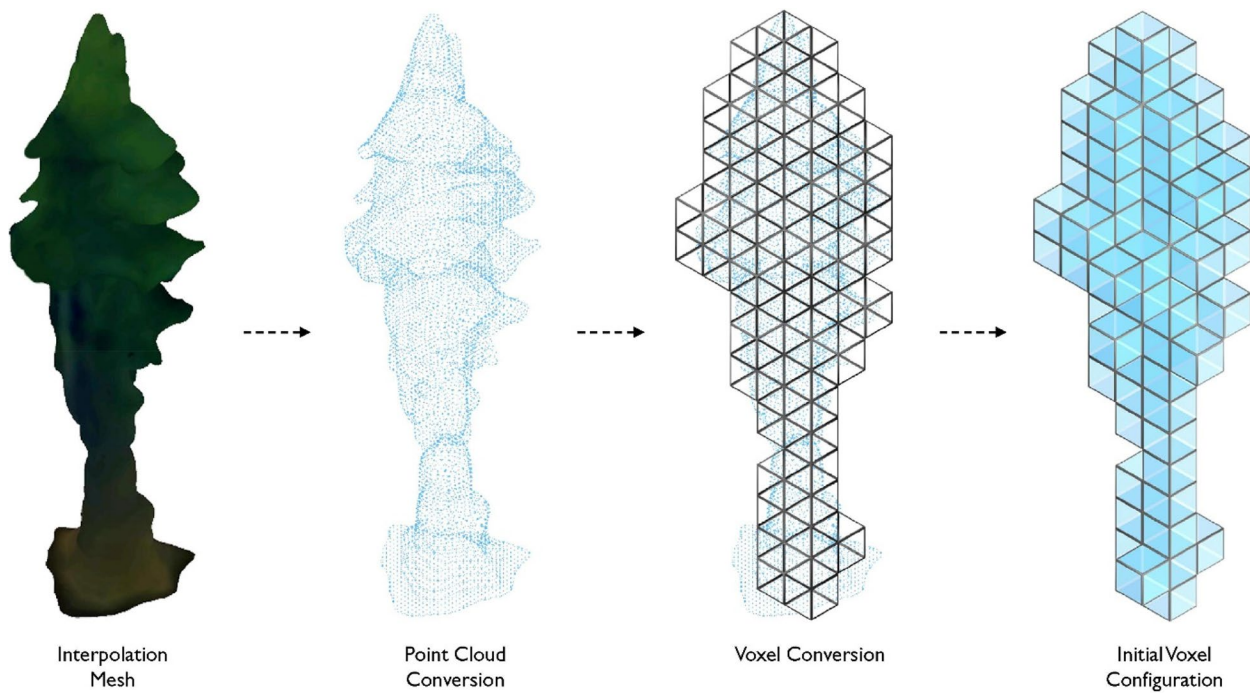
Interpolation   Point Cloud   Voxel Conversion   Initial Voxel
Mesh   Conversion   Configuration

Fig. 6 Steps showing the conversion from the chosen mesh into the initial voxel configuration

Function 2: *TransformMeshToCubicalUnits*

Inputs: baseMesh, unitSizeX, unitSizeY, unitSizeZ, rotateX, rotateY, rotateZ

Output: List of CubeShapes (volumetric units)

Begin

    Determine boundary of baseMesh

    Create a rotation operation based on rotateX, rotateY, rotate

    Apply the rotation to the boundary

    Check for the feasibility of reverse rotation, raise error if not feasible

    Adjust the rotated boundary to align with the original

    Determine the grid dimensions for X, Y, and Z axes

    Initialise a 3D grid to mark filled units

    For each unit in the grid

        Determine its position

        Adjust the position to the rotated system

        If the unit overlaps with the baseMesh

            Mark it as filled in the grid

    Compile a list of positions for filled units

    Initialise a list for CubeShapes

        For each filled unit's position

            Construct a box centered at the position

            Apply the rotation to the box

            Convert the box to a CubeShape and add to the list

    Return the list of CubeShapes

End

The pseudocode describes a process composed of two functions for transforming a 3D mesh into a collection of voxelized cubes. Initially, the mesh undergoes a culling process where only significant parts are retained based on a defined threshold. This reduced mesh is then transformed into a grid of cubic units or voxels. The transformation involves rotating the mesh, adjusting the bounding area, and creating a 3D grid to identify which parts of the space the mesh occupies. Each grid cell representing a part of the mesh is marked, and corresponding cubes are generated at these locations. These cubes, after being rotated back, collectively represent a voxelized version of the original mesh.

4.2 STEP 4- Voxel optimisation

Once the initial voxel conversion is completed, the voxel size and rotation are optimised to minimise the difference between the volume of the input mesh and the volume of the new voxelized geometry. Figure 7 shows multiple results obtained by the single-objective optimisation. Multi-objective optimisation can be run in substitution of it if multiple objectives are needed. The optimisation procedure shown in this paper is obtained by altering six main parameters: voxel dimensions in x, y, and z coordinates, and voxel rotations in x, y, and z coordinates. The difference between the input mesh and the voxel configuration is then calculated for each iteration. The final configuration is then chosen and used for the final step to create the structure and building elements (such as columns, beams, slabs, roofs, facades, etc.).
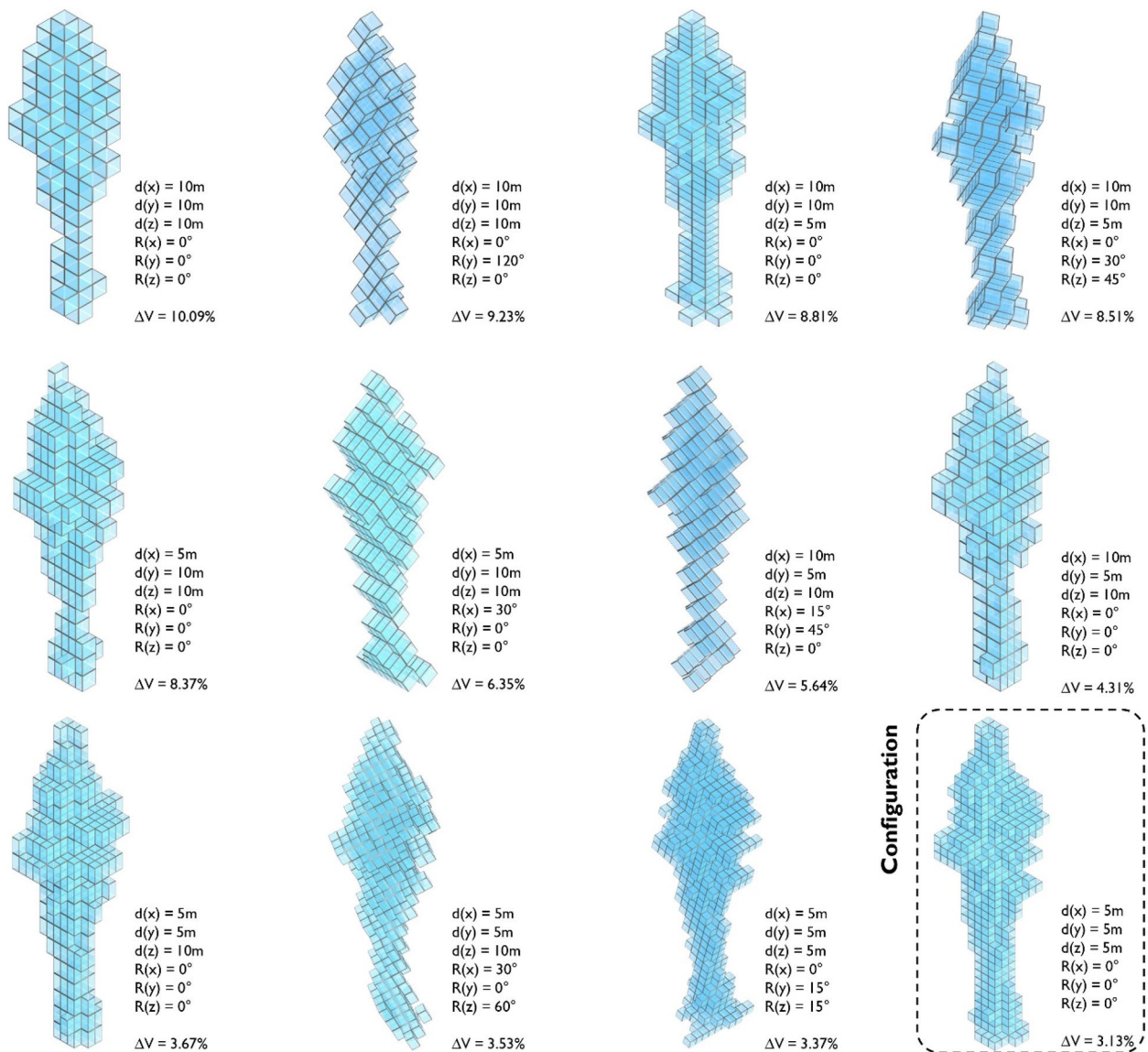
Fig. 7 Multiple voxelized options generated by the single-objective optimisation. The dashed box highlights the final configuration that has been chosen to create the building elements

It is important to highlight the fact that voxel rotations have been included to extend the complexity and versatility of the developed algorithm. Although the voxel rotations generate building structures not suitable for architectural applications in the real world, the opportunity to include rotation angles allows further experimentation in the virtual world – such as generating complex buildings or installation structures for design applications inside the Metaverse. Furthermore, the opportunity to alter voxel angles in addition to voxel dimensions extends the choice of the final configuration to applications not necessarily related to architecture; for instance,

configurations with rotated voxels can be utilised to design furniture structures or other mobile objects.

4.3 STEP 5- Structure generation
The final voxelized volume is chosen amongst the options generated by the single-objective optimisation. Such volume is then converted into structural geometries – columns, beams, and slabs – using custom C# scripts. Figure 8 shows the initial conversion of the chosen voxel configuration into topological elements – vertices and edges – that will be used as reference geometries to generate the structural elements.
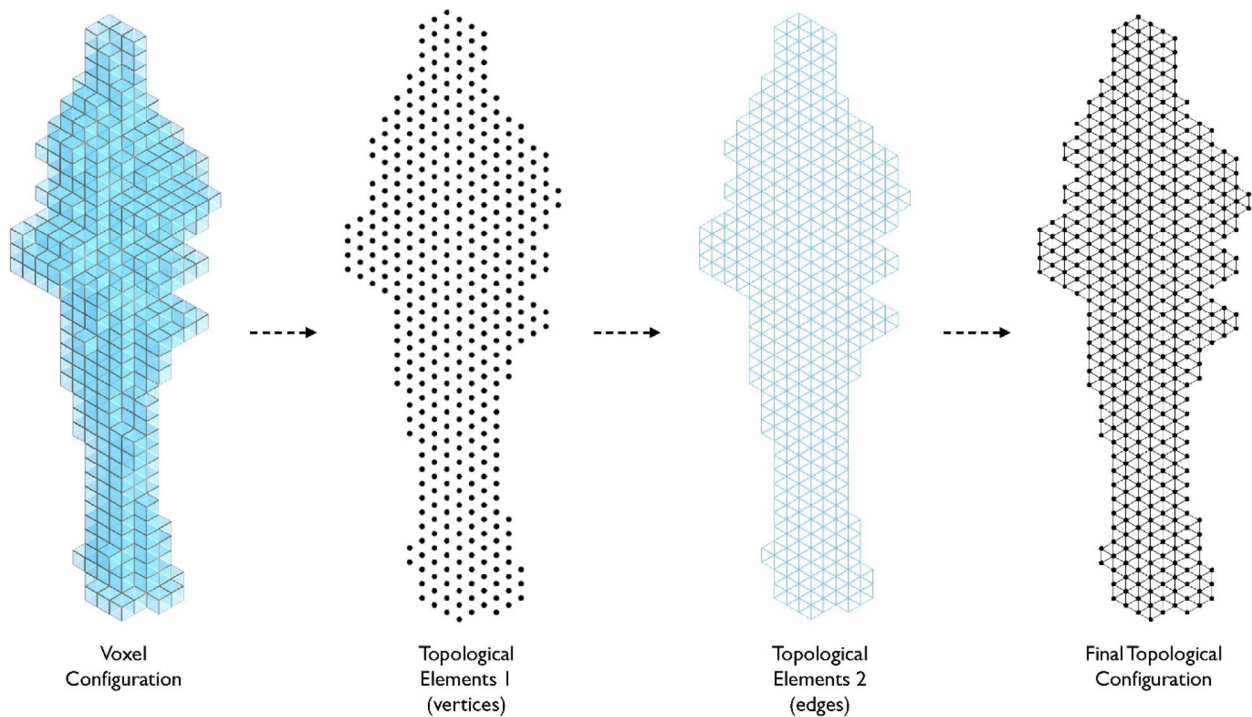
| Voxel Configuration | Topological Elements 1 (vertices) | Topological Elements 2 (edges) | Final Topological Configuration |

Fig. 8 Steps showing the conversion from the final voxel configuration into topological elements

Once the final topological configuration is obtained, the topological elements are converted into structural elements and then analysed using Karamba3D to obtain the structural model and calculate the overall structural utilization (Fig. 9). Further optimisation procedures can be implemented to optimise the elements' size and location.

Once the structural elements are obtained and analysed, the final building configuration is generated by adding the remaining building elements – such as roof surfaces, facade panels, etc.). Figure 10 shows the final building with glazed façade panels. This is only one of the multiple design options that can be achieved with the proposed working pipeline. Multiple design targets can be included to achieve specific design options and configurations.

It is important to highlight the fact that the proposed working pipeline can be used for the design of architectural geometries both in the real and virtual worlds. In fact, the malleability and adaptability of the proposed workflow can be employed both for the conversion of preliminary architectural forms into building structures – which can be analysed and adjusted for construction purposes – and the fast generation of 3D architectural content to populate a specific virtual environment. The possibility of adopting the proposed workflow for designing architecture both in reality and virtually underlies the idea of a seamless architectural aesthetic in the age of artificial intelligence, a time where the rise of *digital anonymity* – namely "the autopoietic condition of digital design, a state in which the combination of decontextualisation and depersonalisation of the design process leads towards emergent and anonymous design results" (Bono et al., 2021) – rectifies the intellectual capability of algorithms and their ability to infer new knowledge by extending certain limits of the human intellect. In doing so, traditional spatial senses and human conceptions are challenged by a new evolutionary process which is translating the independence of human creativity towards the primacy of artificial creativity.

5 Conclusion

This paper has presented a working pipeline to convert AI-generated 3D geometries into building elements. In doing so, tectonics from the latent space have been generated, analysed and then converted into building systems giving life to new and emergent design configurations. The proposed workflow has combined novel AI models and techniques with more traditional computational design procedures, allowing a new experimental methodology in the field of architectural design. The research is currently in its early stages, and the highlighted methodology can be further implemented to obtain a more efficient and integrated approach. To do
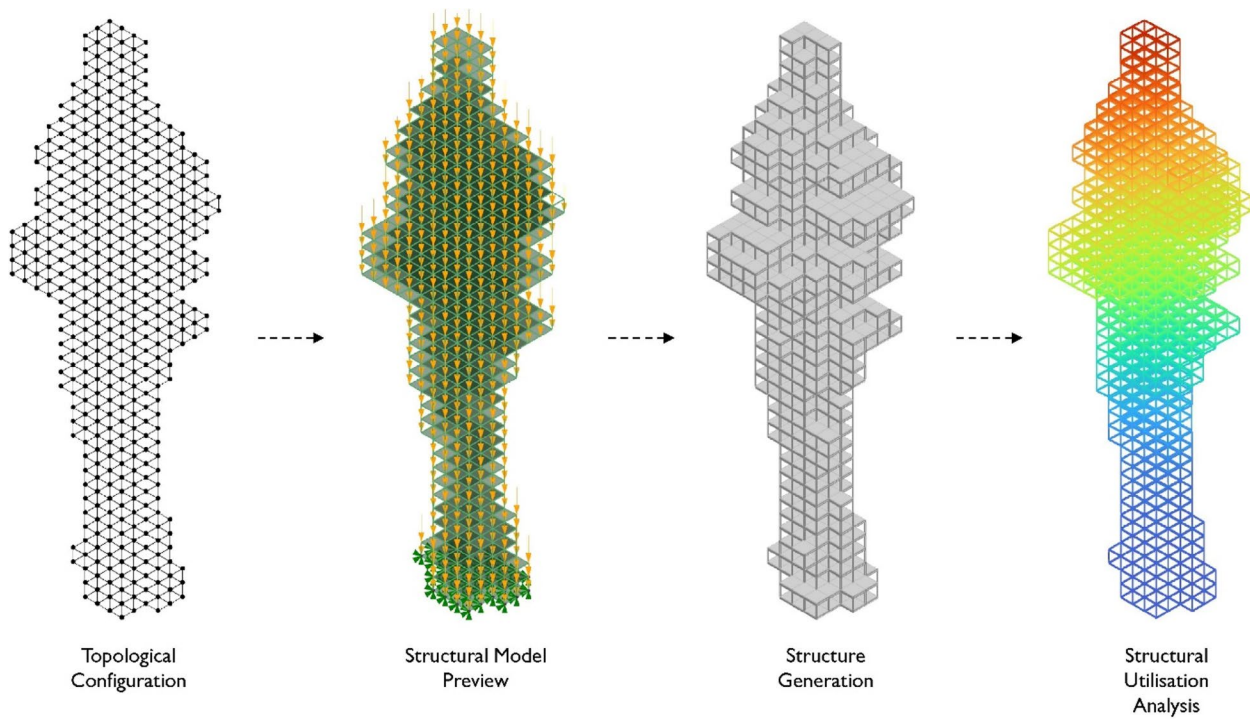
Fig. 9 Steps showing the conversion from the topological configuration into structural elements

Fig. 10 Final building configuration

so, further development is required in two main areas: from one side, a custom AI model would need to be developed and trained to generate more accurate geometries and increase the overall control over the outcomes; from the other side, the current workflow would need to be further implemented towards a custom and standalone application where the overall pipeline is combined into one unique working platform.

The experimentation with tectonics from the latent space has allowed the rise of several considerations regarding the current status of architectural design and the possible ways to develop its conception in the future. First of all, in a time where medical, social, economic, geopolitical, and environmental challenges are changing the way we live within our buildings – for instance, the advent of the COVID-19 pandemic had a significant impact on the architectural design since it has increased the awareness of its stagnation and the need of a significant rethinking of traditional architectural typologies (Gillen et al., 2021) – new design approaches are needed

to create a new architectural space able to translate into reality the needs of the emergent digital society; secondly, the opportunity to deliver more productive and engaging spaces for social exchange, interaction, and communication inside new virtual platforms – such as the Metaverse and the possible work of architects inside such space (Schumacher, 2022) – implies the tendency to expand the democratisation of the design process, namely giving the users the ability to design their own buildings within the new virtual world. For some architects, such a phenomenon represents a detrimental factor for the profession since it undermines the primacy of architects themselves within the design conception; for others, democratising the design process represents an opportunity to expand the profession towards new working activities – such as software development, UX/UI design, visual art, etc. In both cases, the generation of a new architecture for the virtual world can benefit from the use of artificial intelligence inside the design process since it can open a new territory for both topological and typological explorations leading towards a new definition of a possible virtual architecture.

In response to the multifaceted challenges and opportunities delineated, future architecture is poised to undergo a transformative shift. AI generation technology will play a pivotal role, offering innovative solutions that reshape the essence of architectural design. AI's predictive capabilities will enable architects to envision structures that pre-emptively adapt to future scenarios, such as climate change effects or shifting urban demographics. Buildings could dynamically modify their form, function, or environment in real-time, responding to immediate needs like air quality improvement during a health crisis or space reconfiguration in rapidly changing social contexts. Furthermore, AI-driven generative design will push the boundaries of creativity, generating myriad design options based on specified parameters and constraints. This technology can rapidly prototype virtual models, allowing stakeholders to explore and iterate designs in virtual environments. These simulated models will not only be visually represented but also tested for various scenarios, including energy efficiency, structural integrity, and user experience. AI's contribution will extend to the realm of materials as well. Advanced algorithms can aid in discovering new, sustainable materials or optimising existing ones for enhanced performance and reduced environmental impact. This could result in structures that are more energy-efficient, durable, and environmentally friendly. AI in architecture promotes a new era of dynamic, responsive, and sustainable design. Its ability to process complex data and generate innovative solutions will be instrumental in addressing the manifold challenges of our time, crafting a built environment that is adaptable, efficient, and reflective of our evolving sociocultural needs.

Finally, it is worth mentioning some considerations on the creative process generated by using the working pipeline presented in this paper. The idea of combining explorative techniques related to the use of artificial intelligence with more consolidated automated procedures referring to computational design gives life to a generative approach that is not so dissimilar from the idea of "controlled hallucination" explained by Anil Seth (Seth, 2021), particularly concerning building configurations that become reality starting from interpolative and unfamiliar geometries. Referring to the idea of hallucinations as controlled perceptions adopted by the brain to prevent incorrect predictions, Seth states that the brain's predictions are controlled by sensory information from the world which is used to classify such predictions – or hallucinations – and agrees when to convert them into reality. Similarly, building elements obtained from an approximation process of AI-generated 3D geometries – such as the ones shown as results in this paper – represent the controlled representation of tectonics generated inside the latent space, a process created and realised to project them into the real world.

Further considerations can be made also on the relationship between human creativity and machine creativity that arises from the combined use of artificial intelligence and computational design. Starting from the classification of human creativity by Margaret Boden (Boden, 1997) and the classification of machine creativity by Demis Hassabis (Hassabis, 2018), it is interesting to highlight the fact that the proposed pipeline – particularly the studies and tests done during the creation of AI-generated 3D geometries – has allowed to understand a different approach to define creativity in the current age of artificial intelligence. The experimentation with tectonics from the latent space has made clear the fact that creativity is a latent property inside artificial systems. Contrary to both Boden's and Hassabis' classifications – where creativity is evaluated by looking at the final results – the experimentation with latent tectonics has highlighted the fact that creativity is something that lies inside the artificial network rather than in the outcomes that the network can produce. In doing so, the current ways of classifying creativity must be updated towards more open terms of comparison, and new parameters must be considered for evaluating the creative factor.

A possible approach to evaluate machine creativity can be generated by considering its intentional tendency in contrast with the intuitive one promoted by human creativity. The traditional human approach to architectural design is based on the acquisition, sedimentation, and reinvention of knowledge while the new artificial approach

lies in algorithms which can produce endless variations starting from a given set of data. In doing so, the generative capability of algorithms gives life to a new condition in architectural design where the use of artificial intelligence allows the reproduction of autonomous design options. Creativity is becoming an open horizon in which human beings and machines are constantly interacting according to an inbuilt dynamism able to promote new collaborative leadership. For this reason, the future of architectural design will be based on an integrated bottom-up approach where the primacy of human creativity will be maintained inside the definition of the generative inputs, while artificial creativity will be responsible for the definition of the generated outputs: a process acting like a creative loop made of initial decisions and progressive refinements.

Moving in the direction of understanding possible ways to evaluate machine creativity, an interesting example is constituted by three aspects mentioned by Marcus du Sautoy in his book *The Creativity Code*. While explaining the relationship between art and innovation in the age of AI, Du Sautoy suggests three possible parameters to evaluate creativity – novelty, surprise, and value – adding also the fact that nowadays it is quite easy to make something new thanks to the current overabundance of information made available by the hyper-textuality promoted by the virtual world, while surprise and value are more difficult to achieve (Du Sautoy, 2019). If we now try to apply the three Du Sautoy's parameters to evaluate the creative process presented in this paper, one question arises: are tectonics of the latent space new, surprising, and valuable at the same time? If so, the proposed pipeline has shown a methodology to produce authentic forms of creativity; if not, we are still in the limbo of uncertainty and speculation. In both cases, the idea of introducing artificial intelligence inside building design and conception still represents a very slippery territory, but certainly, a field of research that deserves further analysis and investigation.

Declarations

Author details
[1]TP Bennett LLP, London, United Kingdom. [2]Politecnico di Milano, Milan, Italy.

References
Boden, M. (1997). 'Creativity and Artificial Intelligence.' *In Artificial Intelligence. Vol. 103*. Elsevier.

Bono, G., & Guerrieri, P. M. (2021). 'Digital Anonymity Human-Machine Interaction in Architectural Design.' *In Techne. Journal of Technology for Architecture and Environment. Special Series Vol. 2.* Firenze University Press.

Del Campo, M. (2022). *Neural Architecture Architecture and Artificial Intelligence.* ORO Editions.

Du Sautoy, M. (2019). *The Creativity Code: Art and Innovation in the Age of AI* (p. 3) The Belknap Press of Harvard University Press.

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

Galanos, T., & Chronis, A. (2021). 'A Deep-Learning Approach to Real-Time Solar Radiation Prediction.' *In The Routledge Companion to Artificial Intelligence in Architecture*.Routledge.

Gao, J., Chen, W., Xiang, T., Tsang, C. F., Jacobson, A., McGuire, M., & Fidler, S. (2020). *Learning Deformable Tetrahedral Meshes for 3D Reconstruction*.https://arxiv.org/abs/2011.01437.

Gao, J., Shen, T., Wang, Z., Chen, W., Yin, K., Li, D., Litany, O., Gojcic, Z., & Fidler, S. (2022). *GET3D: A Generative Model for High Quality 3D Textured Shapes Learned from Images*.https://arxiv.org/abs/2209.11163.

Gillen, N., Nissen, P., Park, J., Scott, A., Singha, S., Taylor, H., Taylor, I., & Featherstone, S. (2021). *Rethink Design Guide: Architecture for a post-pandemic world*. RIBA Publishing.

Hassabis, D. (2018). *Lecture at the Rothschild Foundation, Waddesdon, Buckinghamshire. 10 October 2018*.https://www.youtube.com/watch?v=d-bvsJW mqlc.

Hawkins, J. (2021). *A Thousand Brain: A New Theory of Intelligence*. Basic Books.

Jun, H., & Nichol, A. (2023). *Shap-E: Generating Conditional 3D Implicit Functions*.https://arxiv.org/abs/2305.02463.

Kasparov, G. (2017). *Deep Thinking: Where Machine Intelligence Ends and Human Creativity Begins* (p. 214). Public Affairs.

Khalid, N. M., Xie, T., Belilovsky, E., & Popa, T. (2022). CLIP-Mesh: Generative Textured Meshes from Text Using Pretrained Image-Text Models. *'22: Proceedings of the SIGGRAPH ASIA 2022.* https://arxiv.org/abs/2203.13333.

Kissinger, H., Schmidt, E., & Huttenlocher, D. (2021). *The Age of Ai: And Our Human Future*. John Murray Publishers.

Ko, J., Ennemoser, B., Yoo, W., Yan, W., & Clayton, M. (2023). 'Architectural Spatial Layout Planning Using Artificial Intelligence.' *In Automation in Construction. Vol. 154*.Elsevier.

Koh, I. (2022). 'Architectural Plasticity. The Aesthetics of Neural Sampling.' In M. Del Campo & N. Leach (Eds.), *Machine Hallucinations: Architecture and Artificial Intelligence. AD Profile 277 (May/June 2022)*. John Wiley & Sons.

Kurzweil, R. (2005). *The Singularity is Near: When Humans Transcend Biology*. Viking.

Lochnicki, G., Kalousdian, N. K., Leder, S., Maierhofer, M., Wood, D., & Menges, A. (2021). 'Co-Designing Material-Robot Construction Behaviors.' *In ACADIA 2021 Realignments: Toward Critical Computation*. Conference Proceedings.

Mildenhall, B., Srinivasan, P., Tancik, M., Barron, J., Ramamoorthi, R., & Ng, R. (2020). *NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis*. https://arxiv.org/abs/2003.08934.

Nichol, A., Jun, H., Dhariwal, P., Mishkin, P., & Chen, M. (2022). *Point-E: A System for Generating 3D Point Clouds from Complex Prompts*. https://arxiv.org/abs/2212.08751.

Prix, W. D., Schmidbaur, K., Bolojan, D., & Baseta, E. (2022). 'The Legacy Sketch Machine.' In D. Del Campo & N. Leach (Eds.), *Machine Hallucinations: Architecture and Artificial Intelligence. AD Profile 277 (May/June 2022)*. Jonh Wiley & Sons.

Schumacher, P. (2022). 'The Metaverse as Opportunity for Architecture and Society: Design Drivers, Core Competencies.' *In Architectural Intelligence*. Vol. 1:11. Springer.

Seth, A. (2021). *Being You: A New Science of Consciousness* (p. 83) Dutton.

Shen, T., Gao, J., Yin, K., Liu, M., & Fidler, S. (2021). Deep Marching Tetrahedra: A Hybrid Representation for High-Resolution 3D Shape Synthesis. *In Advances in Neural Information Processing Systems.* https://arxiv.org/abs/2111.04276.

Xu, D., Jiang, Y., Wang, P., Fan, Z., Shi, H., & Wang, Z. (2022). *SinNeRF: Training Neural Radiance Fields on Complex Scenes from a Single Image*. https://arxiv.org/abs/2204.00928.

Yu, A., Fridovich-Keil, S., Tancik, M., Chen, Q., Recht, B., & Kanazawa, A. (2021). *Plenoxels: Radiance Fields Without Neural Networks*. https://arxiv.org/abs/2112.05131.

Yu, A., Ye, V., Tancik, M., & Kanazawa, A. (2021). *PixelNeRF: Neural Radiance Fields from One or Few Images*. https://arxiv.org/abs/2012.02190.

Publisher's Note