Cybernetics and Intelligence

Research Article

AutoFrit: Bridging rule-based design and industrial generative AI

Yawen Zheng[1,†], Xu Wang[2,†], Fan Dang[3], Pengfei Zhou[4], Yunhao Liu[1,2,✉]

Cite this article: Zheng YW, Wang X, Dang F, et al. *Cybernetics and Intelligence* 2026, **1**(2): 9390007. https://doi.org/10.26599/CAI.2025.9390007

ABSTRACT: The transition to Industry 4.0 is promoting personalized production, challenging traditional design methods reliant on manual expertise and lengthy iterations. Although generative AI shows promise for automation, its industrial use is limited by difficulties in capturing and applying complex rule-based design specifications. To address this, we present "AutoFrit", a comprehensive parametric automobile frit dataset containing design pairs from major manufacturers. Our evaluation shows that models trained on AutoFrit drastically reduce design iteration time from months to minutes while ensuring adherence to industrial standards, effectively addressing the scalability challenges in modern manufacturing.

KEYWORDS: frit design; parametric design; industrial generative AI

1 Introduction

The advent of Industry 4.0 [1] has fundamentally transformed manufacturing requirements, driving enterprises toward personalized production where each product variant demands unique design specifications. This transformation presents unprecedented challenges to traditional design methodologies, as the interpretation and implementation of varying design specifications become increasingly complex and time-consuming. The challenge is particularly acute in parametric design domains, where designers must systematically translate abstract requirements into precise geometric specifications while maintaining compliance with multiple technical constraints.

Our research addresses this challenge through the lens of automobile frit design, as in Fig. 1, which exemplifies the complexity of modern parametric industrial design. Frit patterns, the gradient transitions between opaque and transparent regions in automotive glass, require designers to navigate intricate relationships between parametric specifications and their physical implementation. These patterns serve critical functional purposes: ensuring uniform temperature distribution during heat-forming processes and maintaining structural integrity across different vehicle models. Each manufacturer specifies unique requirements through parametric rules governing pattern density, size gradients, and distribution characteristics, creating a sophisticated design space where multiple constraints must be satisfied simultaneously.

Through our comprehensive field study at a leading global auto glass manufacturer, we identified three fundamental challenges in current design processes:

(i) **Design translation complexity.** Translating parametric specifications into initial design drafts requires designers to manually manipulate thousands of geometric points while managing complex inter-dependencies between design rules. This translation process typically extends over weeks, creating a significant bottleneck in production timelines.

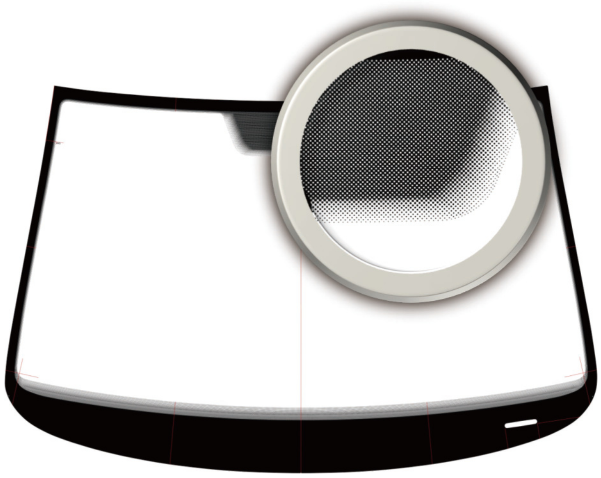


Fig. 1 Frit design example for automobile glass.

(ii) **Parameter optimization burden.** The parameter optimization phase, involving iterative refinement based on manufacturing constraints and client specifications, consumes 1–2 months of the product development cycle.

(iii) **Expertise barrier.** The complexity of implementing parametric rules necessitates months of specialized training before

† Yawen Zheng and Xu Wang contributed equally to this work.

[1] Department of Automation, Tsinghua University, Beijing 100084, China. [2] Global Innovation Exchange, Tsinghua University, Beijing 100084, China. [3] School of Software Engineering, Beijing Jiaotong University, Beijing 100044, China. [4] Department of Informatics and Networked Systems, School of Computing and Information, University of Pittsburgh, Pennsylvania 15260, USA.

✉ Corresponding author. E-mail: yunhao@tsinghua.edu.cn

Received: June 26, 2024; Revised: December 14, 2024; Accepted: March 21, 2025

designers can effectively navigate the design space, creating a substantial barrier to scaling production capacity.

Recently, generative AI [2, 3] has made remarkable progress in automating complex design tasks. Famous platforms like DALL-E [4] and Midjourney [5] have demonstrated the versatility of generative AI in various fields and have become a cutting-edge area of research. To mitigate the paradigm into the industrial engineering area, some researchers have curated domain-specific datasets to train deep learning models [6–8].

While promising, these existing datasets primarily focus on collecting or synthesizing geometric models, without addressing the rule-based specifications. This limitation significantly impacts their utility for personalized and customized industrial design, where adherence to specific parametric rules is essential.

Through extensive analysis of current approaches, we identified two critical gaps hindering the adoption of generative AI in parametric industrial design. First, existing datasets fail to capture the complex mapping between parametric specifications and their visually balanced implementations. Second, current approaches struggle to ensure that generated designs simultaneously satisfy both explicit manufacturing constraints and implicit design requirements.

To address these fundamental challenges and bridge the divide between rule-based parametric design and generative AI, we developed "AutoFrit", a comprehensive system for automated frit pattern design. Our approach combines novel dataset construction methods with advanced deep learning architectures to enable efficient, automated frit design while maintaining strict compliance with manufacturing demands.

Our key contributions are summarized as:

(i) Development of the first large-scale parametric design dataset, containing over 100,000 parametric-frit pairs from ten major automobile manufacturers.

(ii) Implementation of a transformer-based architecture that successfully learns and applies complex parametric design rules while maintaining manufacturing compliance.

(iii) Validation through industrial deployment in a leading auto glass corporation, achieving full-line product coverage and reducing design time from months to minutes.

The remainder of this paper is organized as follows: Section 2 reviews related work in Industry 4.0, parametric design, and industrial generative AI. Section 3 details the construction of "AutoFrit", including our novel pattern synthesis algorithms. Section 4 presents our transformer-based design model and its evaluation results. Section 5 discusses implications and future directions, followed by conclusions in Section 6.

2 Related works

2.1 Industry 4.0

Industry 4.0 [1], characterized by the integration of cyber-physical systems, the Internet of Things (IoT), and advanced data analytics, has significantly influenced design practices across various sectors. Academic research has extensively explored the implications of these technologies on design, highlighting the shift towards smart, interconnected, and highly efficient systems. Key studies have examined the role of IoT in enabling real-time data exchange and monitoring, which enhances the design and development of products by providing immediate feedback and facilitating iterative improvements. For instance, research [9] has elucidated how smart factories leverage IoT and cyber-physical systems to optimize production processes and enhance customization capabilities. Advanced data analytics and artificial intelligence play crucial roles in predictive maintenance, quality control, and supply chain management, as illustrated in Ref. [10], emphasizing the transformation of design processes into more adaptive and data-driven practices. Moreover, additive manufacturing [11] or 3D printing [12], is a pivotal component of Industry 4.0, as it allows for rapid prototyping and bespoke production, fundamentally altering traditional design methodologies. Studies have also investigated the challenges associated with implementing Industry 4.0 technologies, including data security, interoperability, and the need for new skill sets among designers. Despite these challenges, the ongoing integration of Industry 4.0 technologies continues to foster innovation and efficiency, driving the evolution of design practices toward more intelligent and responsive systems.

2.2 Parametric design

Parametric design [13] has been extensively explored in academic literature, showcasing its transformative impact on architecture, engineering, and various design disciplines [14–17]. Early theoretical foundations were laid by researchers such as George Stiny and James Gips with their work on shape grammars [18, 19], which formalized the use of rules and parameters in design generation. A Language Pattern [20] further contributed to this framework by emphasizing systematic approaches to design through patterns. Modern parametric design is often facilitated by software tools like Grasshopper for Rhino [21] and Dynamo for Revit [22], which enable the creation of complex geometries through algorithmic processes. Studies [23] exemplify the application of these tools in pushing architectural boundaries. In urban design, Michael Batty's research [24] on urban modeling and simulation demonstrates the use of parametric tools in optimizing urban environments. Additionally, parametric design is applied in industrial design, fashion, and engineering, as evidenced by the customizable product designs by companies like Nike and the innovative garments by fashion designers such as Iris van Herpen. Despite its advancements, parametric design faces challenges, including the complexity of managing large datasets and the steep learning curve associated with parametric software. However, ongoing research and the integration of artificial intelligence and machine learning promise to address these challenges and further enhance the capabilities of parametric design systems.

2.3 Related parametric design dataset

Other parametric design datasets have been collected for training and evaluation benchmarks for generative AI models.

ShapeNet [6] is a richly annotated, large-scale repository of 3D CAD shapes. The shapes are organized under the WordNet taxonomy. ShapeNet covers 55 common object categories with about 51,300 unique 3D models.

UIUC aerofoil coordinates database [7] includes over 1600 airfoils, which cover a wide range of applications from low Reynolds number airfoils for UAVs and model aircraft to jet transports and wind turbines.

Ship-hull design dataset [8] developed a large ship dataset containing 52,591 design variations of existing classes of ships, approximately 5000 different hull forms from FORMDATA [25] series and remaining hulls from synthetic variations based on the parametric approach [26].

2.4 Visual balance

Visual balance [27] has been a focal point of academic research within the fields of art, design, and human–computer interaction, exploring its critical role in aesthetics and user experience.

Theoretical foundations of visual balance can be traced back to Gestalt psychology [28], which emphasizes the human perception of patterns and the principles of organization, including balance, symmetry, and equilibrium. Reference [29] has extensively studied visual perception and aesthetics, which outlines how balance contributes to the overall harmony and stability of a visual composition. Contemporary studies [30] have extended these theories to digital interfaces, where visual balance significantly affects usability and user satisfaction. For instance, research [31] has demonstrated that balanced design elements enhance the aesthetic appeal and perceived usability of websites. In graphic design and layout, tools like grid systems and the golden ratio are often employed to achieve visual balance, ensuring that elements are proportionally and harmoniously arranged. Moreover, advancements in computational design [32, 33] have facilitated the automatic assessment and optimization of visual balance in digital content, as explored in studies involving algorithmic and machine-learning approaches to design evaluation. Despite these advancements, challenges remain in quantitatively measuring visual balance [34–36] and its subjective perception across diverse user groups. Nonetheless, ongoing research continues to refine the understanding of visual balance, emphasizing its importance in creating visually pleasing and functionally effective designs.

2.5 Generative AI for design

Generative AI for design [37] has emerged as a pivotal area of academic research, revolutionizing the creative processes in architecture, product design, and digital media. Early explorations into generative algorithms laid the groundwork, with John Holland's development of genetic algorithms [38] demonstrating how evolution-inspired methods could optimize complex design problems. In recent years, advancements in machine learning, particularly deep learning, have significantly enhanced the capabilities of generative design tools. Generative adversarial networks (GANs) [39] have been instrumental, allowing for the creation of highly realistic and novel design outputs. Applications of GANs and other AI models in design are extensive, ranging from automating architectural layout generation to creating intricate fashion patterns and personalized product designs. Academic study [40] showcases the integration of AI in developing adaptive and responsive architectural structures. Additionally, tools like DALL-E [4] and Midjourney [5] have broadened the scope of generative AI, enabling the synthesis of imaginative and artistic visuals from textual descriptions. Despite the remarkable progress, challenges such as ensuring the originality and ethical use of AI-generated designs persist. Ongoing research is focused on improving the interpretability of generative models, enhancing their integration into existing design workflows, and addressing issues related to intellectual property and authorship. As generative AI continues to evolve, it promises to redefine the boundaries of creativity and efficiency in design disciplines.

3 Constructing AutoFrit

The construction of AutoFrit follows a three-stage process that progressively translates manufacturer design requirements into optimized frit patterns. We first collect candidate patterns from major automobile manufacturers (typical examples can be seen from Fig. 2), focusing on their descriptive design constraints rather than specific implementations. These constraints, ranging from contour conformity to pattern interlock requirements, form the foundation of our dataset. The collected constraints are then transformed into initial pattern designs through a rule-based pre-meshing process, which implements basic geometric arrangements following explicit design rules. Finally, we refine these initial patterns using dynamic relaxation, where descriptive constraints are mapped to specific optimization parameters for fine-grained pattern adjustment.

3.1 Collecting candidate patterns

Generally, windshield patterns can be categorized into black borders, third sun visors, and bottom frits. The black border is a band-like gradient at the top and sides of the glass. The third sun visor around the rearview mirror prevents sunlight through gaps in the car's roof or foldable sun visors. Bottom frits function similarly to black borders, but with wider transition zones and a greater black-to-white gradient span. Due to their different functions and coverage areas, design requirements for these patterns vary.

We have undertaken the collection of frit pattern design data from over ten prominent international and domestic automobile manufacturers, including but not limited to Volkswagen, Audi, and BMW. Unlike conventional pixel-based visual datasets, which often treat pixels as independent units, frit pattern elements exhibit substantial correlations and dependencies. This intrinsic interconnectedness implies that relying solely on the production blueprints used by technicians could result in several challenges, such as data redundancy, irrational data structures, and sparse data distribution.

To mitigate these issues, our initial phase focused on gathering representative design codes and extracting generalized features rather than amassing large quantities of raw data.

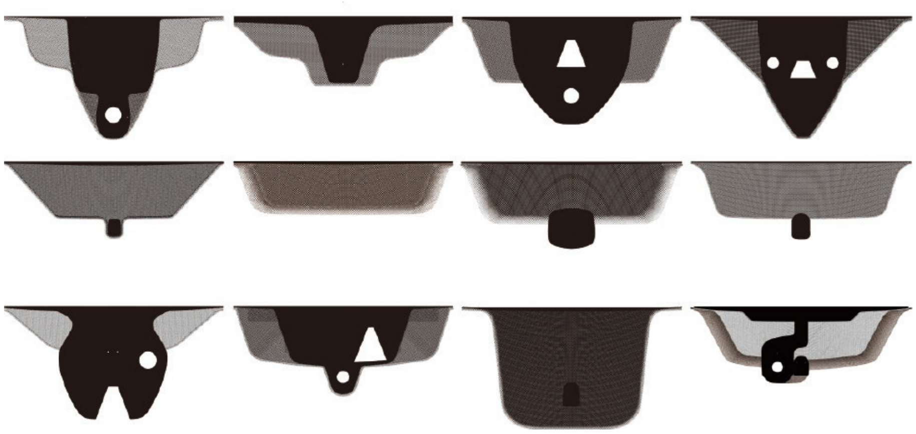


Fig. 2 Typical frit patterns of the third sun visor.

To effectively manage diverse design specifications, we organize the dataset using a bipartite graph structure that captures the relationship between patterns and their associated constraints. The dataset comprises:

- **Design patterns (*D*)** containing template regions where patterns will be generated.
- **Constraints (*C*)** encompassing the descriptive requirements for each region.
- **Relationships (*R*):** defining the associations between patterns and their applicable constraints.

As is shown in Fig. 3, this structured organization enables systematic pattern generation in subsequent stages while maintaining the integrity of manufacturer-specific requirements.

3.2 Parametric pattern generation

To emphasize the interconnected nature between design patterns and design codes, we developed a rule-based parametric design method to preliminarily expand the dataset. This method allows for a systematic and scalable approach to data augmentation, thereby enhancing the comprehensiveness and utility of the dataset for subsequent analysis and application.

By linking various design patterns with constraint rules, we facilitate the use of modular rule-based algorithms in parametric design, allowing for easier combination and manipulation. However, it is important to note that the combination of multiple design objectives often results in an over-constrained problem. To achieve an overall harmonious pattern, it is crucial to address the multi-objective optimization problem. This requires sophisticated techniques to balance and reconcile the competing constraints and objectives, ensuring a cohesive and aesthetically pleasing design outcome.

As shown in Table 1, the mapping between descriptive constraints and dynamic relaxation parameters establishes a systematic framework for translating manufacturer requirements into optimizable parameters. Each descriptive constraint maps to specific geometric controls in our optimization framework. For instance, contour conformity requirements are implemented through a combination of boundary separation distances, region control boundaries, and contour alignment angles. Similarly, pattern alignment specifications are achieved through the control of angular relationships and uniform spacing parameters. This mapping enables our dynamic relaxation algorithm to fine-tune

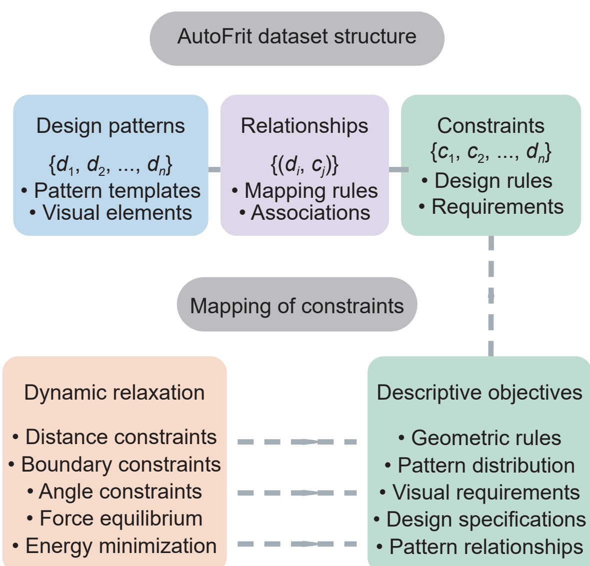


Fig. 3 Dataset structure.

Table 1 From descriptive constraints to optimizable parameters

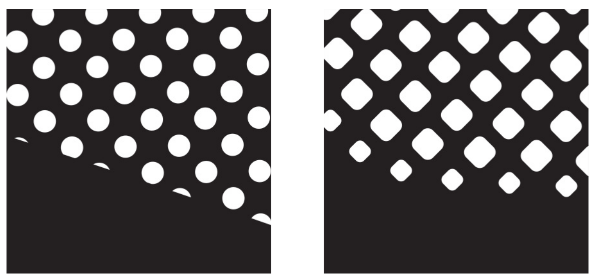
Constraint	Dynamic relaxation optimization
Contour	Distance (boundary separation), Boundary (region control), Angle (contour alignment)
Alignment	Angle (diagonal/omni-directional), Distance (uniform spacing)
Gradient	Distance (spacing variation), Boundary (transition zones)
Interlock	Distance (pattern spacing), Angle (orientation)

pattern arrangements while maintaining compliance with the original design intent.

Rule-based pre-meshing. We have identified several common design objectives for third sun visor frit patterns. According to these requirements, we performed a preliminary rule-based meshing of the specified two-dimensional contours of the car window glass. In this mesh, the vertices represent the center positions of each pattern. The distance between the mesh vertices, along with the pattern size and distribution density, collectively determine the overall layout and spacing of the frit patterns. This approach allows for a structured and systematic arrangement of patterns, ensuring that design objectives are met effectively.

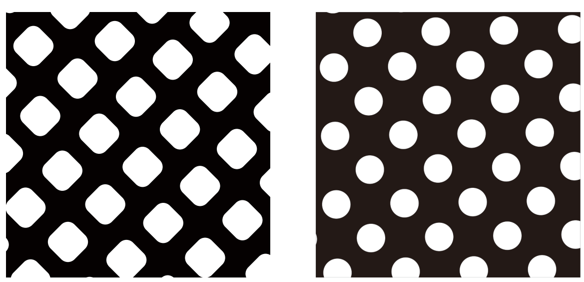
Contour. For the frit patterns in the third sun visor area, the intricate patterns must be strictly confined within the contours. Otherwise, there is a risk of obstructing the driver's normal field of vision. As in Fig. 4, some automobile manufacturers opt to directly cut the frit patterns along the outer contour, while others require the patterns to follow the contour direction at the boundary, thus preserving their integrity. This distinction reflects varying design preferences and safety considerations among different manufacturers.

Alignment. Upon the concept of contour confinement, frit patterns in the third sun visor area exhibit two typical alignment approaches, as illustrated in Fig. 5. One approach involves aligning the frit pattern diagonally while maintaining contour conformation on both sides. The other approach requires alignment in each horizontal, vertical, and diagonal direction. It is evident that, under the condition of constant spacing between arrays, the latter, more stringent alignment criteria, combined



(a) Cut by contour (b) Conform to contour

Fig. 4 Frit patterns along contours.



(a) Diagonal (b) Omni-directional

Fig. 5 Frit pattern alignment.

with contour conformation, may result in an over-constrained problem.

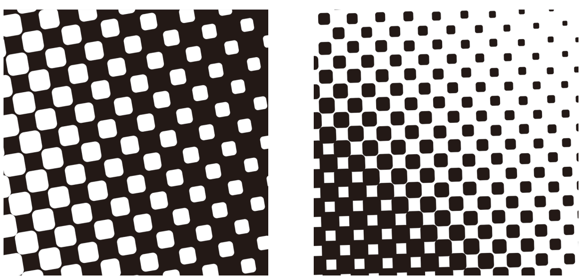
Consequently, technicians often adjust the spacing between arrays near the contour boundary to achieve two objectives: creating aesthetically pleasing transitional effects and maintaining the uniformity of internal arrays.

Gradient. For the third sun visor area, on the one hand, the diameter of the frit patterns gradually decreases from the center of the windshield outward, seamlessly integrating the pattern with the opaque strip at the upper edge of the windshield and the surrounding areas, creating a harmonious appearance. On the other hand, the spacing of the frit pattern array also changes uniformly, balancing the conflict between contour confinement and strict alignment. This dual approach ensures both aesthetic integration and functional precision.

Besides, the frit patterns in the black border area serve to bridge the black ceramic coating and the transparent glass, thereby preventing optical distortion defects during the glass hot-bending process. At the bottom of the front windshield, where the curvature of the glass is greater, the area and gradient of the frit pattern transition are correspondingly larger, as is shown in Fig. 6.

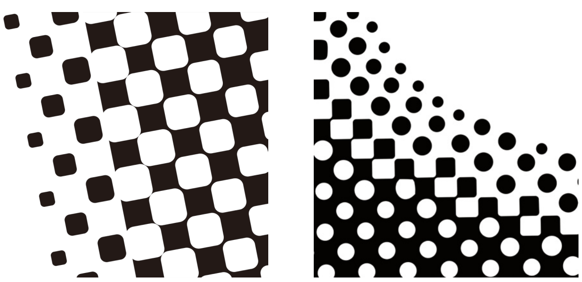
Interlock. Frit patterns can be categorized into two types: positive patterns and negative patterns, as is shown in Fig. 7. The positive consists of black ceramic designs on transparent glass, while the negative are hollow transparent designs on the base of black ceramic coatings. For functional and aesthetic considerations, many automotive manufacturers, such as Volkswagen and Audi, have developed unique glass frit patterns. These patterns have evolved from simple black dots to a combination of positive and negative designs, creating a dynamic and fluid transition effect and enhancing the visual appeal of the frit patterns.

Constrained dynamic relaxation. Dynamic relaxation is a numerical method employed for solving force equilibrium geometrical structures, capable of characterizing the dynamic behavior of physical systems discretized over time. Its fundamental approach involves assigning a certain mass to the nodes of interest and defining force relationships among nodes



(a) Bottom frits (b) Third sun visor

Fig. 6 Frit pattern gradient.



(a) Bottom frits (b) Third sun visor

Fig. 7 Frit pattern interlock.

related to stiffness. Under the influence of loads, points oscillate around the equilibrium state. Based on the accumulated momentum of forces, the method iteratively updates the positions of points by a certain step size until convergence to the equilibrium state (or a local solution with minimal overall energy). Dynamic relaxation finds common application in the fields of structural mechanics and architectural design, providing a solution strategy for computational mechanics based on particle systems.

We used pre-meshed elements as initial variables, with the regions of interest (ROI) determined based on the geometric boundary relationships outlined in the blueprints to prevent excessively slow convergence. Utilizing Delaunay triangulation, dependency relationships are defined between local patterns. Optimization objectives such as center point distances, line angle intersections, and collisions with boundary lines are established to generate constrained manifolds. Dynamic relaxation and projection dynamics are employed to adaptively adjust the quantity, position, and size of patterns, aiming to achieve a visually uniform distribution while ensuring force equilibrium and minimal potential energy states are reached.

Exact ROI. With the boundary contour C shifted, different types of grids are generated according to the subdivision regions. This structured approach ensures a precise definition and computation of the regions of interest, maintaining the integrity and logical consistency of the grid generation process based on geometric boundaries.

Mesh triangulation. Through triangulation, we establish the positional dependencies between neighboring grid vertices.

This structured approach ensures a precise definition and

Algorithm 1 Exact region of interest

Require: Closed boundary contour C; Original grid vertex set $\{v_k\}$

Ensure: Edge grid vertex set V_{ROI}; Internal grid vertex set V_{in}

Calculate the offset distance d as

$$d = \alpha\sqrt{d_{\mathrm{v}}^2 + d_{\mathrm{h}}^2}$$

where $\alpha \geq 1$, d_{v} is the vertical spacing and d_{h} is the horizontal spacing of the grid array;

Offset C by distance d and get contour C'

Define R as the annular regions between the original contour C and the offset contour C'; R' is the internal regions of the offset contour C'

Extract subsets of $\{v_k\}$ that fall within R and R':

$$V_{\mathrm{ROI}} \leftarrow \{v_k \in R\}$$
$$V_{\mathrm{in}} \leftarrow \{v_k \in R'\}$$

Algorithm 2 Delaunay triangulation

Require: Edge grid vertex set V_{ROI}; Internal grid vertex set V_{in}

Ensure: Edge grid edge set E_{ROI}; Internal grid edge set E_{in}

```
Triangulate all vertices and get original edge set {e_i}
for each edge e_i in {e_i} do
    if both endpoints of e_i belong to V_in then
        Add e_i to E_in
    else
        Add e_i to E_ROI
    end if
end for
```

computation of the positional dependencies between grid vertices, maintaining the integrity and logical consistency of the grid generation process based on Delaunay triangulation.

Dynamic relaxation. Based on dynamic relaxation (force-based) and projective dynamics (potential energy-based) methods, influence relationships are established between each flower point and its surrounding points, defined by distance constraints, boundary constraints, and optionally angle constraints, as is illustrated in Fig. 8

1. **Distance constraints:** For an edge e_{ab} with vertices v_a and v_b at its endpoints ($e_{ab} \in E_{\mathrm{ROI}}$ or $e_{ab} \in E_{\mathrm{in}}$), the constraint is $\|e_{ab}\| = d$ or $\|e_{ab}\| \geq d$ or $\|e_{ab}\| \leq d$, where the value of d is determined by the type of e_{ab} and the values of d_{v} and d_{h}.

For example, for $e_{ab} \in E_{\mathrm{in}}$, as singularities are almost non-existent within this point set, it can be specified that $\|e_{ab}\| = \|e_{ab}\|_0$, where $\|e_{ab}\|_0$ is the initial length of e_{ab} after mesh triangulation.

2. **Boundary constraints:** For any region S_i enclosed by a closed boundary ∂S_i and a set of vertices $\{v_k\}$, the constraint is $\forall v_k, v_k \notin S_i$ (the point lies outside the curve boundary) or $\forall v_k, v_k \in S_i$ (the point lies within the curve boundary).

For example, for $\{v_k\} \subseteq V_{\mathrm{in}}$, if the region S satisfies $C' = \partial S$, it can be specified that $\forall v_k, v_k \in S$.

3. **Angle constraints:** For any edge e_i, the horizontal reference line e_{h}, and the vertical reference line e_{v}, where θ_{h} is the angle between e_i and e_{h}, θ_{v} is the angle between e_i and e_{v}, and α_i is a constant angle parameter, the numerical conditions that θ_{h}, θ_{v}, and α_i need to satisfy can be specified similarly to the distance constraints.

4. **Optimization objective:** For the set of constraints $\{g_i\}$ and their corresponding weight set $\{w_i\}$, the optimization objective is $\min\{\sum_{i=1}^{N} w_i g_i\}$.

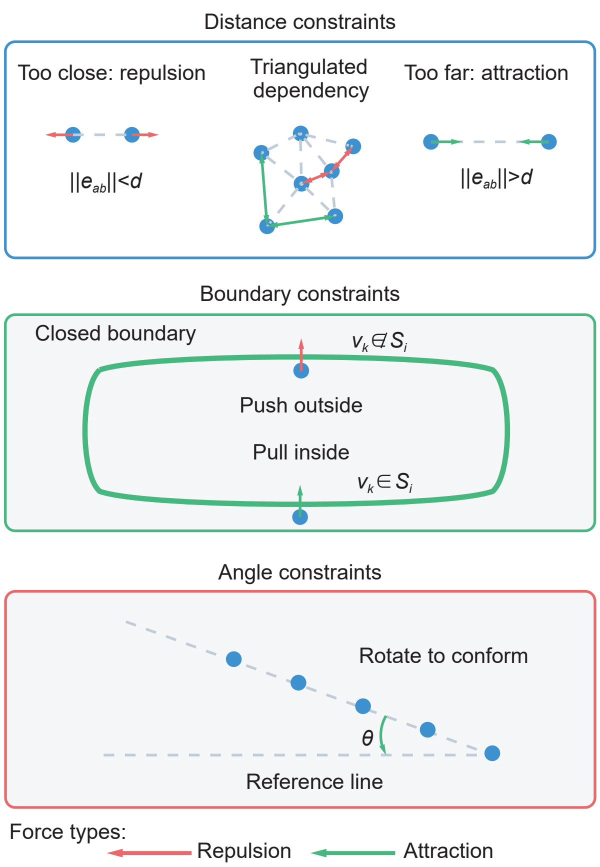


Fig. 8 Dynamic relaxation.

With the iteration step length, maximum iteration count, and termination conditions defined, the final positions of the frit patterns can be iteratively generated.

Adjust the size of each pattern according to its position. For an arbitrary point v_k, let the Delaunay triangle edges with v_k as an endpoint be $\{e_k\}$, and their lengths be

$$D_k = \sum_{k=1}^{n} \|e_k\|$$

The radius r_k of the pattern corresponding to v_k satisfies

$$r_k = \frac{D_k}{D_0} r_0$$

where r_0 is the size parameter of the frit pattern, and D_0 is determined by the values of d_v and d_h.

3.3 Crowd-sourced visual fine-tuning

AutoFrit presents an innovative framework for embedding rule-based frit pattern design algorithms within a collaborative software engineering environment, specifically engineered for frontline frit-design personnel. As illustrated in Fig. 9, the framework implements a comprehensive workflow that encompasses configuration loading, area defining, pattern designing, parameter adapting, geometry engineering, and quality validating processes.

Configuration loading. The workflow initiates with users importing predefined configuration files containing essential parameters and design settings. These configuration files define the initial conditions for the frit pattern design process. The modular nature of these files enables users to modify specific parameters to align with their design requirements while maintaining system consistency.

Area defining. In this phase, the framework leverages integration with mainstream CAD systems to facilitate precise area selection. Users can define regions of interest through an intuitive CAD interface, utilizing reference points for accurate spatial positioning. This integration significantly reduces technical barriers by allowing designers to work within familiar CAD environments while accessing advanced pattern-generation capabilities.

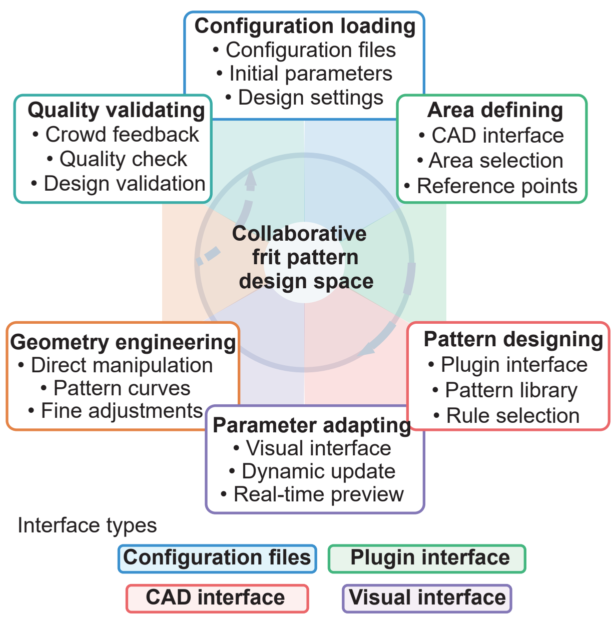


Fig. 9 Crowd-sourced pattern design workflow.

Pattern designing. The design implementation phase introduces a sophisticated plugin interface that provides access to an extensive pattern library. Users can select from predefined design rules or create custom codes through a standardized interface. This approach ensures both flexibility in design choices and consistency in pattern generation while maintaining the technical rigor required for industrial applications.

Parameter adapting. Parameter fine-tuning is facilitated through a visual interface that enables dynamic updates with real-time preview capabilities. This interface incorporates intuitive controls for adjusting design parameters, allowing users to modify pattern attributes such as density, thickness, and curvature while immediately visualizing the results. The real-time feedback mechanism significantly accelerates the design-to-manufacturing pipeline by reducing iteration cycles.

Geometry engineering. This phase enables direct control and manipulation of pattern curves, supporting fine adjustments through an advanced interface. This capability allows designers to leverage their experiential judgment while maintaining algorithmic precision. The system provides immediate visual feedback during manipulation, ensuring that adjustments align with both aesthetic requirements and functional specifications.

Quality validating. The workflow culminates in an expert review phase that implements crowdsourcing mechanisms for quality assurance. This phase includes systematic quality checks and design validation procedures, ensuring that patterns meet both technical specifications and aesthetic standards. The crowdsourcing approach democratizes the fine-tuning process while maintaining rigorous quality control.

Drawing on principles from the field of human–computer interaction (HCI), this framework exemplifies user-centered design and participatory design theories. User-centered design ensures that the tool meets the actual needs of pattern-makers by providing an intuitive interface that reduces the learning curve and increases productivity. Participatory design theory is reflected in the crowdsourcing aspect of the tool, enabling users to contribute to the refinement of design algorithms through their interactions with the software.

By harnessing the power of crowdsourcing, we gather and refine design requirements from real-world applications, enabling manual fine-tuning of rule-based algorithms. This not only aligns the algorithms more closely with practical design needs but also democratizes the fine-tuning process, allowing pattern-makers to impose a direct influence through the tool's development progress. Our methodology accelerates the workflow of frit-design professionals, allowing them to produce high-quality designs more efficiently.

Most importantly, the implementation supports enterprise-level scalability requirements through streamlined workflow integration, efficient pattern creation mechanisms, robust quality maintenance protocols, and collaborative fine-tuning capabilities. Through this comprehensive approach, the framework successfully accelerates the workflow of frit-design professionals while maintaining high-quality design standards. The system's architecture supports scalable manufacturing requirements while reducing technical barriers through intuitive interfaces and automated validation processes.

4 Application: Transformer-based automobile frit design

AutoFrit is valuable for training generative AI models. In this section, we train a transformer-based automobile frit design model and show its performance with a double-blind evaluation. Beyond this model, AutoFrit could be easily integrated with other applications, we will discuss its potential in the next section.

Generative AI has made significant strides in learning from existing designs and creating new ones. In this section, we conceptualize frit design as a conditional language translation process. Specifically, we use explicit local rules as conditional constraints, border lines as the input sentence, and the arrangement of frits as the output sentence. An encoder–decoder transformer model is employed to learn the mapping from constraints and input sequences to output sequences.

In the model design, a transformer is employed to build the generative AI design model, leveraging its significant performance in natural language processing. To handle the constraints described in Section 3.2 and the input border point sequences, we utilize two embedding layers for vectorization. The top-left corner is used as the start token, and the bottom-right corner as the end token, to delineate the prediction sequences. We employ traditional multi-head attention to capture key dependencies between sequences.

To accelerate the training process, we introduce a group-based token compression method. Since each design contains thousands of points, directly predicting each point with an auto-regressive decoder would result in poor computational efficiency. Observing that points in the center area are generally distributed evenly along a line, we propose a group-based token compression method to reduce the number of output tokens. Specifically, we convert the design into a mesh using Delaunay triangulation and search for connected points that form a line from the top right to the bottom left. We retain only the key pivot points to represent each line. Figure 10 shows the model architecture. We utilize the multiple transformer encoders to convert the curve points to the hidden representations, and multiple transformer decoder to generate the predicted pivot points. The curve points are uniformly sampled from the boundary. Our model is implemented on Pytorch [41]. The layer number of both encoders and decoders, the hidden dimension, and the head number of multi-head attentions are the hyper-parameters to be tuned. In our experiments, we set the layer number as 3, the head number as 8, the dimension as 64. Figure 11 shows the loss of our training process, which indicates the convergence.

Since there are no direct criteria for aesthetic design, we developed a double-blind evaluation method to compare the performance of different approaches, including purely manual, parametric design, and generative AI design.

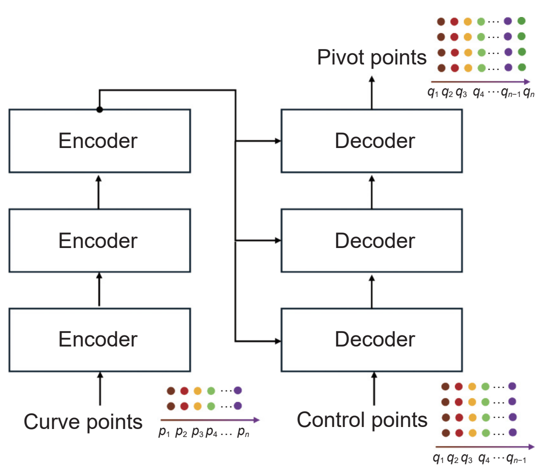


Fig. 10 Transformer model architecture with the layer number as 3.

清華大学出版社 Tsinghua University Press | Available on IEEE Xplore

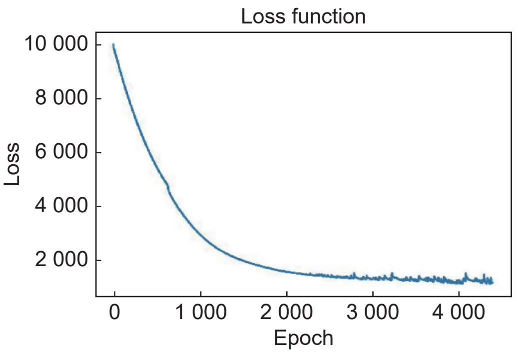


Fig. 11 The loss of our model.

We assembled a team of ten experts to score the design results from these different methods.

The selection criteria for review experts are as follows: (i) Experts must have extensive experience in designing frit patterns for automotive windshields and a record of outstanding achievements in this field. (ii) Experts must possess a minimum of ten years of professional experience. (iii) Experts should represent diverse regions and manufacturing departments. Based on these criteria, we established a pool of experts and selected the top ten individuals as the final review panel. During the evaluation phase, various methods were applied to create frit pattern designs for windshields of different automotive brands. These design files were anonymized and distributed to the ten experts for scoring.

The experts assessed each design based on visual aesthetic standards, scoring within a range of 0 to 10 points. To ensure fairness in the evaluation process, a five-level scoring standard was implemented:

- Level 1 (0–2 points): The design completely fails to meet customer aesthetic expectations.
- Level 2 (2–4 points): The design exhibits some gaps in meeting customer aesthetic expectations.
- Level 3 (4–6 points): The design aligns reasonably well with customer aesthetic expectations.
- Level 4 (6–8 points): The design significantly exceeds customer expectations.
- Level 5 (8–10 points): The design fully meets or even surpasses customer aesthetic expectations.

Each expert was required to submit their evaluations within a single day, following these standards. The scores for various designs and methods were then aggregated through basic statistical analysis to determine the final results.

The evaluation scores are summarized in Table 2. The results show that our generative AI method outperforms all other methods. Figure 12 shows a case of Volkswagen products. In this case, the generative model achieved results slightly superior to manual designs and significantly outperformed those produced through parametric design. While the parametric design results appeared more structured in localized features, they lacked overall fluidity and coherence.

Table 2 The evaluation result

	Manual	Parametric	Generative
avg	7.80	5.20	7.82
min	7.20	4.00	7.30
max	9.00	5.50	9.00

After dozens of iterations, the model was deployed in a global auto glass company, achieving a 98.13% coverage rate across products. Compared to traditional manual methods, feedback from the company indicates that the model now completes new designs in minutes.

5 Discussion and future work

Our future work has two goals.

5.1 Completing AutoFrit

The current dataset only covers about ten typical vehicle models, leaving many models uncovered. The project aims to expand this data to cover over 50% of mainstream vehicle models. Currently, each vehicle model in the dataset has only around 10,000 design drawings. We plan to increase the number of design drawings for each category tenfold to enhance the accuracy of neural network training.

5.2 Exploiting AutoFrit

We expect AutoFrit to become a significant advancement in industrial design research. Potential applications are envisioned include:

- **Automobile windshield design**. AutoFrit serves as a crucial resource for designing automobile windshields. The dataset helps in creating aesthetically pleasing frit patterns that ensure a smooth transition between the opaque border and the transparent glass, thus minimizing optical distortion and enhancing visual appeal.
- **Optimization of manufacturing processes**. The dataset aids in optimizing manufacturing processes, particularly in the furnace heat-forming stage. By utilizing detailed frit patterns, manufacturers can achieve a more uniform temperature distribution, reducing defects like optical distortions during the production of automobile glass.
- **Training generative AI models**. Beyond transformer-based models, the AutoFrit dataset is invaluable for training various generative AI models such as generative adversarial networks (GANs) and variational autoencoders (VAEs). These models can generate innovative frit patterns that are both functional and aesthetically pleasing, advancing the development of AI-driven design tools.

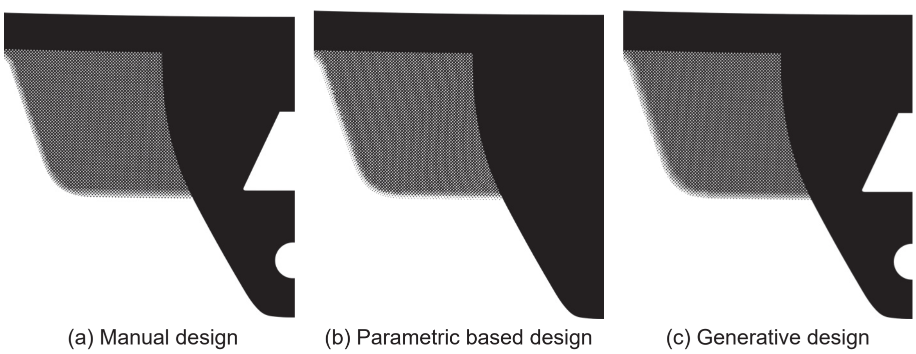


Fig. 12 A case study for automobile frit design.

● **Benchmark for visual balance research**. AutoFrit can be employed as a benchmark dataset for visual balance research. The intricate patterns in the dataset allow researchers to develop and test new algorithms aimed at achieving optimal visual balance, which is crucial in fields like graphic design and human–computer interaction.

6 Conclusions

In this work, we develop a large automobile frit design dataset covering over ten automobile manufacturers. To show its potential, we propose transformer-based generative AI models to capture the spatial dependencies and visual-balanced features. Evaluation by double-blind evaluation shows our model outperforms traditional manual design and parametric design. We also deploy the model in a worldwide auto glass company, covering one in every five cars in the world and two in every five cars in China. The dataset will benefit applications like automobile windshield design, optimization of manufacturing processes, training generative AI models, and benchmarking for visual balance research.

Acknowledgements

This work is supported in part by the National Natural Science Foundation of China under Grant Nos. 62232004, 62302259, 62532009 and Tsinghua University–Fuzhou Joint Institute for Data Technology.

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

References

[1] Zhou, K.; Liu, T.; Zhou, L. Industry 4.0: Towards future industrial opportunities and challenges. In: Proceedings of the 12th International Conference on Fuzzy Systems and Knowledge Discovery, 2147–2152, 2016.

[2] Stokel-Walker, C.; Van Noorden, R. The promise and peril of generative AI. *Nature* Vol. 614, No. 7947, 214–216, 2023.

[3] Zhang, B.; Zhu, J.; Su, H. Toward the third generation artificial intelligence. *Science China Information Sciences* Vol. 66, No. 2, Article No. 121101, 2023.

[4] OpenAI. DALL·E 2. 2022. Available at https://openai.com/index/dall-e-2/

[5] Midjourney Inc.. Midjourney. 2024. Available at https://www.midjourney.com

[6] Chang, A. X.; Funkhouser, T.; Guibas, L.; Hanrahan, P.; Huang, Q.; Li, Z.; Savarese, S.; Savva, M.; Song, S.; Su, H.; et al. ShapeNet: An information-rich 3D model repository. *arXiv preprint* arXiv:1512.03012, 2015.

[7] UIUC Applied Aerodynamics Group. UIUC airfoil coordinates database. 2024. Available at https://m-selig.ae.illinois.edu/ads/coord_database.html

[8] Khan, S.; Goucher-Lambert, K.; Kostas, K.; Kaklis, P. ShipHullGAN: A generic parametric modeller for ship hull design using deep convolutional generative model. *Computer Methods in Applied Mechanics and Engineering* Vol. 411, Article No. 116051, 2023.

[9] Kagermann, H.; Wahlster, W. Ten years of Industrie 4.0. *Sci* Vol. 4, No. 3, Article No. 26, 2022.

[10] Lee, J.; Ni, J.; Singh, J.; Jiang, B.; Azamfar, M.; Feng, J. Intelligent maintenance systems and predictive manufacturing. *Journal of Manufacturing Science and Engineering* Vol. 142, No. 11, Article No. 110805, 2020.

[11] Wong, K. V.; Hernandez, A. A review of additive manufacturing. *International Scholarly Research Notices* Vol. 2012, No. 1, Article No. 208760, 2012.

[12] Shahrubudin, N.; Lee, T. C.; Ramlan, R. An overview on 3D printing technology: Technological, materials, and applications. *Procedia Manufacturing* Vol. 35, 1286–1296, 2019.

[13] Monedero, J. Parametric design: A review and some experiences. *Automation in Construction* Vol. 9, No. 4, 369–377, 2000.

[14] Jabi, W. *Parametric Design for Architecture*. London, UK: Routledge, 2013.

[15] Rohm, T.; Jones, C. L.; Tucker, S. S.; Jensen, C. G. Parametric engineering design tools and applications. In: Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, 657–664, 2020.

[16] Gane, V. Parametric design: A paradigm shift? S.M. Thesis. Massachusetts Institute of Technology, 2004.

[17] Wu, J.; Li, T.; Liu, X.; Wang, L. Optimal kinematic design of a 2-DOF planar parallel manipulator. *Tsinghua Science and Technology* Vol. 12, No. 3, 269–275, 2007.

[18] Stiny, G.; Gips, J. Shape grammars and the generative specification of painting and sculpture. In: Proceedings of the IFIP Congress, 1971.

[19] Stiny, G. Introduction to shape and shape grammars. *Environment and Planning B: Planning and Design* Vol. 7, No. 3, 343–351, 1980.

[20] Alexander, C.; Ishikawa, S.; Silverstein, M.; Jacobson, M.; Fiksdahl-King, I.; Shlomo, A. *A Pattern Language: Towns, Buildings, Construction*. Oxford University Press, 1977.

[21] Davidson, S. Grasshopper. 2024. Available at https://www.grasshopper3d.com/

[22] Autodesk, Inc. Dynamo. 2024. Available at https://dynamobim.org/

[23] Schumacher, P. Parametricism: A new global style for architecture and urban design. *Architectural Design* Vol. 79, No. 4, 14–23, 2009.

[24] Batty, M. *Urban Modelling*. Cambridge University Press, 1976.

[25] Guldhammer, H. E. FORMDATA1: Some systematically varied ship forms and their hydrostatic data. *International Shipbuilding Progress* Vol. 9, 108–133, 1962.

[26] Khan, S.; Kaklis, P. From regional sensitivity to intra-sensitivity for parametric analysis of free-form shapes: Application to ship design. *Advanced Engineering Informatics* Vol. 49, Article No. 101314, 2021.

[27] Mokarian, M. A. Visual balance in engineering design for aesthetic value. Master Thesis. University of Saskatchewan, 2007.

[28] Köhler, W. Gestalt psychology. *Psychologische Forschung* Vol. 31, No. 1, XVIII–XXX, 1967.

[29] Arnheim, R. *Art and Visual Perception: A Psychology of the Creative Eye*. Berkeley: University of California Press, 1954.

[30] Jahanian, A.; Vishwanathan, S. V. N.; Allebach, J. P. Learning visual balance from large-scale datasets of aesthetically highly rated images. In: Proceedings of the SPIE 9394, Human Vision and Electronic Imaging XX, 93940Y, 2015.

[31] Lavie, T.; Tractinsky, N. Assessing dimensions of perceived visual aesthetics of web sites. *International Journal of Human-Computer Studies* Vol. 60, No. 3, 269–298, 2004.

[32] Menges, A.; Ahlquist, S. *Computational Design Thinking*. New York, USA: John Wiley & Sons, 2011.

[33] Coros, S.; Thomaszewski, B.; Noris, G.; Sueda, S.; Forberg, M.; Sumner, R. W.; Matusik, W.; Bickel, B. Computational design of mechanical characters. *ACM Transactions on Graphics* Vol. 32, No. 4, Article No. 83, 2013.

[34] Lok, S.; Feiner, S.; Ngai, G. Evaluation of visual balance for automated layout. In: Proceedings of the 9th International Conference on Intelligent User Interfaces, 101–108, 2004.

[35] Lopes, D.; Correia, J.; Machado, P. Towards the automatic evaluation of visual balance for graphic design posters. 2023. Available at https://computationalcreativity.net/iccc23/papers/ICCC-2023_paper_51.pdf

[36] Wang, Y.; Zhang, Y. The Influence of visual balance of interface elements layout on the attractiveness of APP guide pages. In: *Intelligent Human Systems Integration (IHSI 2023): Integrating People and Intelligent Systems. AHFE (2023) International Conference. AHFE Open Access, Vol. 69*. Ahram, T.; Karwowski, W.; Di Bucchianico, P.; Taiar, R.; Casarotto, L.; Costa, P. Eds. AHFE International, USA, 2023.

[37] Thoring, K.; Huettemann, S.; Mueller, R. M. The augmented designer: A research agenda for generative AI-enabled design. *Proceedings of the Design Society* Vol. 3, 3345–3354, 2023.

[38] Holland, J. H. Genetic algorithms. *Scientific American* Vol. 267, No. 1, 66–72, 1992.

[39] Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial networks. *Communications of the ACM* Vol. 63, No. 11, 139–144, 2020.

[40] Menges, A.; Kannenberg, F.; Zechmeister, C. Computational co-design of fibrous architecture. *Architectural Intelligence* Vol. 1, No. 1, Article No. 6, 2022.

[41] Ketkar, N.; Moolayil, J. Introduction to PyTorch. In: Deep Learning with Python. Apress, 27–91, 2021.