A Survey of Large Language Models for Vector Graphics Generation

Authors: Sun Yifei, Zeng Guokun, Zeng Guokun

Date: 2026-01-08T23:17:25+00:00

Abstract

[Objective] To systematically review the research progress of large language models in the field of vector graphics generation, and to clarify their technological evolution paths and core research issues. [Methods] By retrieving relevant literature from the past five years, existing studies are classified and compared according to technological paradigms, and a comprehensive analysis is conducted from the perspectives of semantic modeling approaches, levels of geometric representation, and generation frameworks. [Results] The study shows that this field has undergone three stages: from vision–language semantic guidance, to diffusion-model-assisted vector generation, and then to structured SVG generation with large language models as the core. During this evolution, generation quality, structural consistency, and editability have been progressively improved. [Limitations] Current research still exhibits shortcomings in geometric precision control, the construction of a unified evaluation system, and stability in complex design scenarios, and the related methods rely heavily on high-quality data and computational resources. [Conclusion] Large language models provide a new research paradigm for vector graphics generation, but their potential in structured modeling and human–computer collaborative design remains to be further explored.

Full Text

Preamble

A Survey of Large Language Models for Vector Graphics Generation

Sun Yifei[1], Kuo-Kun Tseng[2]
[1]Harbin Institute of Technology (Shenzhen), Shenzhen 518000, China

Abstract:
[Objective] This paper systematically reviews the research progress of large language models in vector graphics generation, clarifying its technological evolution

path and core research issues. [**Methods**] By searching relevant literature from the past five years, existing studies are classified and compared according to technical paradigms, with comprehensive analysis conducted from perspectives of semantic modeling approaches, geometric representation levels, and generation frameworks. [**Results**] Research indicates that the field has experienced three stages: visual-language semantic guidance, diffusion model-assisted vector generation, and structured SVG generation centered on large language models, with gradual improvements in generation quality, structural consistency, and editability. [**Limitations**] Existing research still faces deficiencies in geometric precision control, construction of unified evaluation systems, and stability in complex design scenarios, with relevant methods heavily dependent on high-quality data and computational resources. [**Conclusions**] Large language models provide a new research paradigm for vector graphics generation, yet their potential in structured modeling and human-computer collaborative design remains to be further explored.

Keywords: large language model; vector graphics; SVG generation
Classification Number: TP391.41

In the digital age, visual content has become central to information transmission and user experience. Vector graphics, due to their resolution independence, editability, and lightweight nature, play an indispensable role in web design, UI/UX, animation, illustration, and data visualization [1,2]. However, traditional vector graphics design workflows are often time-consuming and labor-intensive, demanding high professional skills from designers, including deep understanding and proficient operation of geometric shapes, paths, colors, layer structures, and underlying SVG code [3,4]. This leads to two major industry pain points: low design efficiency and high professional barriers, severely limiting creative freedom for non-professional users and junior designers.

In recent years, the rapid advancement of artificial intelligence technology, particularly the rise of large language models, has brought unprecedented opportunities to address these challenges. Representative studies such as LLM4SVG [5], StarVector [6], and SVGFusion [7] have achieved breakthrough progress in complex semantic modeling, cross-modal joint input, and high-precision commercial integration, strongly demonstrating the enormous potential of this direction. LLMs, with their powerful natural language understanding, generation, and reasoning capabilities, are gradually penetrating various domains, including computer vision, multimodal interaction, and even creative design [8–10]. By transforming natural language instructions into structured vector graphics code, LLMs promise to simplify design workflows, lower professional skill barriers, and enable ordinary users to generate high-quality vector graphics through simple text descriptions. This paradigm shift not only enhances design efficiency but also stimulates broader creative expression.

This survey aims to systematically organize and analyze the research progress, core technologies, practical applications, and future trends and challenges of large language models in vector graphics generation. We will delve into how key

technologies such as multimodal learning, semantic understanding, and human-computer collaborative design empower LLMs to learn from multi-source information including text and images to generate complex vector graphics. Through summarization of existing literature, this paper aims to provide researchers in this field with a comprehensive perspective and offer valuable insights for future research directions. Specifically, this paper will focus on how to leverage the advantages of LLMs to overcome challenges in vector graphics generation regarding geometric precision, editability, diversity, and semantic consistency, ultimately exploring research frontiers in text-to-SVG generation.

2 Literature Search and Research Methods

This study aims to comprehensively review research progress of large language models in vector graphics generation; therefore, the design of literature search strategy is crucial. We employed a series of keyword combinations, including "Large Language Models (LLMs)," "Vector Graphics," "SVG Generation," "Multimodal Learning," "Human-Computer Co-design," "Text-to-Vector," and "Image-to-Vector," conducting extensive searches in major academic databases and conference proceedings. The search timeframe primarily focused on the past five years to ensure coverage of the latest research achievements, while appropriately tracing some foundational works in this field.

The research methodology of this survey mainly includes the following aspects: (1) **Classification and Induction:** Selected literature is divided into different categories based on research content and technical routes, such as diffusion model-based methods, Transformer architecture-based methods, multimodal fusion methods, and methods focusing on specific application scenarios. This classification clearly reveals the technological development trajectory and main research directions in this field. (2) **Comparison and Analysis:** In-depth comparison and analysis of technical details, advantages, and limitations of different methods. For example, comparing different models' performance in generation quality, editability, diversity, computational efficiency, and complex semantic understanding capability, focusing not only on quantitative performance metrics but also on underlying innovative ideas and technical principles. (3) **Problem and Challenge Identification:** Based on analysis of existing research, identifying key problems and challenges currently faced by large language models in vector graphics generation, such as insufficient geometric precision, difficulty in ensuring semantic-visual consistency, lack of large-scale high-quality datasets, and complexity of human-computer interaction. (4) **Trend and Outlook Prediction:** Combining current development trends in artificial intelligence and computer graphics, forecasting future development directions and potential research hotspots for large language models in vector graphics generation, including more powerful multimodal understanding capabilities, finer control granularity, more efficient generation, and broader application scenarios.

Additionally, during the writing and organization of this survey, we reasonably utilized large language models and other AI tools to assist with preliminary liter-

Figure 1

Figure 1: Figure 1

ature classification, language polishing, and logical structure checking, without affecting academic judgment and content selection. All research viewpoints, literature selection, and technical judgments were independently completed by the authors and verified based on original paper content. This approach aims to improve survey writing efficiency and expression accuracy, rather than replacing the academic research process itself.

Through the above research methodology, this survey strives to provide a comprehensive, in-depth, and insightful perspective to offer valuable references and guidance for further research in this field.

3.1 Development Stage Review

This paper systematically reviews the technological evolution since 2023 by combining representative research and systematic applications. As shown in Figure 1

, related research has roughly experienced four stages: early capability exploration, specialized model development, multimodal fusion methods, and commercial integration, with research focus and technical paradigms gradually shifting.

Figure 1 Timeline of Large Language Model SVG Generation (2023–2025)

The first stage shown in Figure 1 primarily relies on the embedding alignment capability of "vision-language"models for text semantics, treating vector graphics generation as a continuous optimization problem. Representative methods such as CLIPDraw maximize the similarity between text embeddings and rendered results in a shared semantic space through iterative optimization of vector path parameters. The core principle of these methods lies in constructing cross-modal alignment loss functions, but due to reliance on rasterized rendering as an intermediate representation, they struggle to directly constrain SVG structural rationality and geometric precision.

The second stage introduces diffusion models as strong visual priors, transforming the vector generation process into a denoising optimization problem constrained by text conditions. Methods represented by DiffSketcher and VectorFusion typically employ Score Distillation Sampling (SDS) loss to distill semantic gradients from pretrained text-to-image diffusion models for updating Bézier curve or path control points. The key idea of this stage is leveraging diffusion models'distribution modeling capability to enhance visual quality, but geometric control still mostly occurs in pixel space, limiting SVG structure interpretability and editability.

With the enhancement of LLMs' reasoning and structure generation capabilities, research has gradually entered a stage centered on language models. This stage treats SVG as a structured program representation, utilizing Transformers to model SVG code sequences, or learning continuous representations of vector graphics through variational autoencoders and latent space diffusion models. Related work demonstrates that by explicitly modeling path hierarchies, graphic composition logic, and semantic planning processes, LLMs can significantly improve structural consistency and editability while maintaining visual quality. The core transformation of this stage lies in moving from "image-intermediated vector generation" to "program structure-centered vector modeling." To clearly present the technical evolution logic of this field, Table 1 summarizes the specific differences in core paradigms, representative works, and advantages/disadvantages across the three stages.

Table 1 Comparison of Technical Paradigm Evolution

Stage	Core Paradigm	Representative Works	Advantages	Limitations
Stage 1: Vision-Language Guidance	Cross-modal alignment	CLIPDraw [11], StyleCLIPDraw [12]	No large training data required, style diversity	Slow optimization, difficulty ensuring complex geometric structure rationality
Stage 2: Diffusion Model Assistance	Score Distillation Sampling (SDS)	DiffSketcher [16], VectorFusion [17]	High visual quality, rich details, texture expression	Geometric control still based on pixel space, limited SVG structure editability
Stage 3: LLM-Centric Generation	Serialization/Program generation	StarVector [6], Chat2SVG [28], SVGBuilder [43]	Strong structural consistency, supports semantic editing, high code interpretability	High computational cost, geometric precision fine-tuning (e.g., Bézier curve parameters) still challenging

Figure 2

Figure 2: Figure 2

3.2 Technical Paradigm Classification

However, temporal dimension alone cannot fully reveal the technical essence behind various methods. To further clarify technical paradigm differences in existing research, this paper constructs a classification system for large language models in vector graphics generation, as shown in Figure 2

. The system categorizes existing methods into three main paradigms: optimization-based methods, diffusion-assisted models, and autoregressive/sequence methods centered on large language models. This classification helps more clearly understand the applicability and technical boundaries of different models in downstream tasks such as sketch generation, style transfer, and high-fidelity rendering.

Figure 2 Classification System of Large Language Models for Vector Graphics Generation

4 Application Practice and Collaborative Mechanisms

The rapid development of large language models in vector graphics generation extends beyond theoretical research, demonstrating enormous potential in practical applications and human-computer collaborative design. By integrating LLMs into design tools, automated workflows, and multimodal interactive interfaces, design efficiency can be significantly improved, professional barriers lowered, and creative expression boundaries expanded.

4.1 Automated Design and Content Generation

The most direct application of LLMs lies in automatically generating various vector graphics content to accelerate design workflows. For example, Figma Config 2025 [FIGURE:2025] and CorelDRAW 2025' s AI-driven features embody LLM empowerment of commercial design tools. Figma' s dynamic vector generation function supports real-time semantic editing, meaning users can quickly adjust graphic attributes and layouts through natural language instructions without manually manipulating complex Bézier curves or layers [41]. CorelDRAW' s PowerTRACE algorithm combined with LLM semantic understanding significantly improves the accuracy and speed of bitmap-to-vector conversion, solving problems of insufficient precision and excessive time consumption in traditional vectorization tools when handling complex images.

At the academic research level, many works also focus on automating generation of specific vector graphics types. IconShop [42] utilizes autoregressive Transformers to achieve text-guided vector icon synthesis by tokenizing SVG paths and text descriptions, enabling unconditional and text-conditioned icon generation.

This model surpasses existing methods in generation quality, diversity, and flexibility, supporting applications such as icon editing, interpolation, semantic composition, and automated design suggestions. SVGBuilder [43] introduces a component-based autoregressive model for efficient colored SVG generation, achieving generation speeds hundreds of times faster than traditional methods and surpassing state-of-the-art models in both efficiency and quality through training on the ColorSVG-100K dataset.

DiffPlanner [44] proposes a direct vector floor plan generation method using Transformer-based conditional diffusion models, avoiding the conversion process from vector data to raster and back to vector, thereby reducing complexity and information loss. This aligns with designers' iterative design processes, capable of handling complex vector data and generating high-quality floor plans. Cui et al. (2025) [45] and Chen et al. (2025) [46] explore LLM applications in generating visualization items and data visualization, revealing LLM potential in information visualization through automated visualization item generation and NL2VIS (Natural Language to Visualization) benchmarks. These studies demonstrate that LLMs can automatically generate various vector graphics from simple icons to complex floor plans and data visualizations, greatly expanding the boundaries of automated design. Table 2 further systematically organizes mainstream models discussed in this chapter, comparing their characteristics from dimensions such as core architecture, supported modalities, and task types to showcase the diversity of the current technological ecosystem.

Table 2 Horizontal Evaluation of Mainstream Model Characteristics and Capabilities

Model	Core Architecture	Supported Modalities	Task Type	Features/Innovations
IconShop [42]	Transformer	Text	Icon generation	Supports automated design suggestions and semantic composition
StarVector [6]	Multimodal LLM	Image + Text	Cross-modal SVG generation	Supports joint image and text input, multimodal alignment
SVGBuilder [43]	Component-based Transformer	Text	Colored SVG generation	Fast generation speed, introduces component-based generation
Chat2SVG [28]	LLM + Diffusion Model	Text + Image	Interactive SVG editing	"What you say is what you get" interactive editing, low barrier

Model	Core Architecture	Supported Modalities	Task Type	Features/Innovations
DiffPlanner [44]	Transformer	Text	Floor plan generation	Direct vector floor plan generation without rasterization intermediate

4.2 Multimodal Interaction and Human-Computer Collaborative Design

The combination of LLMs and vector graphics involves not only automated generation but more importantly promotes a new paradigm of multimodal interaction and human-computer collaborative design. Users can interact with the system through multiple modalities including natural language, sketches, and reference images to jointly complete design tasks.

Chat2SVG is a typical example, allowing users to intuitively edit generated vector graphics through natural language instructions. This "what you say is what you get" interaction method enables non-professional users to easily modify graphic attributes such as color, shape, and layout, greatly lowering the editing threshold. SketchAgent [47] is a language-driven sequential sketch generation method that leverages off-the-shelf LLMs, enabling users to create, modify, and refine sketches through conversational interaction without model training. This method processes string-based actions as vector graphics, capturing the dynamic nature of sketching and enabling diverse sketch generation and human-computer collaborative drawing. StarVector, as a multimodal LLM, supports joint image and text input, meaning users can provide a sketch or reference image combined with natural language descriptions to generate precise SVG code. This multimodal input approach better aligns with designers' actual workflows and can more accurately capture user intent. The VectorPainter framework proposed by Hu et al. (2024) [48] achieves advanced stylized vector graphics synthesis through reference image-guided text-to-vector generation, converting reference image pixels to vector strokes and then optimizing stroke positions and colors according to text. This method enables users to customize generation results by providing style references, further enhancing the flexibility of human-computer collaboration.

In more complex scenarios, LLMs can also serve as intelligent assistants to help designers with concept exploration and creative inspiration. Vinker et al. (2023) [49] proposed a method for decomposing visual concepts into hierarchical tree structures using large vision-language models, where each node represents a subconcept. This enables designers to explore new concepts, conduct infinite visual sampling, and combine different aspects of trees to generate novel visual ideas, thereby performing creative exploration under natural language sentence guidance. This mechanism transforms LLMs' knowledge and reasoning capabilities

into designers' creative tools, greatly expanding design possibilities.

Additionally, LLMs demonstrate potential in understanding and generating domain-specific graphics. Lee et al. (2025) [50] studied LLM applications in generating mathematical diagrams, automating the generation of mathematical prompt diagrams in SVG format and evaluating their quality and effective prompting strategies. This has significant implications for education and scientific visualization. Wang et al. (2025) [51] explored multimodal learning methods in PUML diagram assistant optimization, aiming to improve drawing efficiency and accuracy and promote intelligent development of drawing tools.

Overall, LLM application practices in vector graphics generation are evolving from simple automated generation to more advanced multimodal interaction and human-computer collaborative design. By providing more natural and intuitive interaction methods, LLMs are empowering broader user groups to participate in creative design and providing professional designers with powerful intelligent assistance tools, jointly driving innovation in the design field.

4.3 Vector Graphics Editing and Conversion

Beyond generating vector graphics from scratch, LLMs are also being applied to vector graphics editing and conversion tasks, further enhancing the flexibility and efficiency of design workflows. These applications typically involve understanding, modifying, and converting existing vector graphics between different formats.

In vector graphics editing, Kucha et al. (2025) [52] constructed a large-scale instruction-guided vector image editing dataset VectorEdits, containing over 270,000 SVG image-instruction pairs for training and evaluating models that modify vector graphics based on text commands. Preliminary experiments indicate that LLMs still face challenges in accurate and effective editing, highlighting the difficulty of this task. Nevertheless, the release of this dataset will greatly promote research on natural language-driven vector graphics editing. Warner et al. (2023) [53] proposed VST (Vector Style Transfer), a novel design tool implementing flexible style transfer between vector graphics. Designers can adjust cross-design element correspondences and customize style attributes, providing potential application directions for LLMs in stylized editing.

In vector graphics conversion, raster image vectorization is a long-standing challenge. Ma et al. (2022) proposed the LIVE model, a deep learning-based raster-to-SVG conversion model that can maintain image topology and generate compact SVGs with hierarchical structure and consistent semantics. SuperSVG [54] is a superpixel-based model that significantly improves reconstruction accuracy and inference time by decomposing images into superpixels and adopting a two-stage self-training framework. These vectorization techniques can be combined with LLMs to achieve more intelligent image-to-SVG conversion; for example, LLMs can guide the vectorization process according to user instructions to generate specific styles or structures.

Furthermore, LLMs can be used to convert vector graphics to other forms or extract vector information from other forms. Zhang et al. (2023) proposed a motion vectorization pipeline that converts motion graphics videos into SVG motion programs, enabling higher-level editing. This allows designers to create animation variants by adjusting timing, motion, and appearance. This ability to abstract video content into editable vector programs opens new avenues for LLM applications in animation and motion graphics design.

At a more macro level, the capabilities of multimodal large models in understanding and processing different modality data also provide a foundation for cross-modal conversion of vector graphics. For example, the two-stage framework of Research on LLM-Based Vector Art Generation, although primarily focusing on text-to-vector, can be extended in its semantic parsing stage to handle other modality inputs. StarVector directly supports joint image and text input, achieving text-image-vector cross-modal conversion. These capabilities enable LLMs to serve as a central hub connecting different design input and output forms, implementing more flexible and intelligent design workflows.

In summary, LLMs demonstrate enormous application potential in vector graphics editing and conversion. By combining advanced deep learning techniques, LLMs can not only automatically generate vector graphics but also understand, modify, and convert existing graphics, thereby providing designers with more comprehensive and intelligent assistance tools. Future research will continue exploring how to improve LLM accuracy, efficiency, and flexibility in these tasks to meet increasingly complex design requirements.

5 Existing Problems and Challenges

Despite significant progress in vector graphics generation using large language models, the field still faces numerous complex and profound challenges spanning technical, data, evaluation, and ethical dimensions. Deep understanding of these issues is crucial for advancing future research.

5.1 Challenges in Geometric Precision and Path Quality

The essence of vector graphics lies in their precise geometric representation, such as Bézier curve control points, path connection methods, shape fills and strokes. LLMs often face insufficient precision when generating these low-level geometric details. Zhang et al. (2024) noted that existing text-to-vector (T2V) methods lack geometric constraints, resulting in poor path quality. While LLMs excel at processing text sequences, precisely mapping high-level semantic concepts to complex geometric paths requires profound understanding of spatial relationships, topological structures, and aesthetic principles. For example, generating a "circle" is simple, but generating a "precise, smooth circle that seamlessly connects with background elements" requires fine control over parameters such as coordinates, radius, and stroke width.

Additionally, generated SVG code may contain syntax errors or unreasonable

structures, causing graphics to render incorrectly or be difficult to edit. Timofeenko et al. (2024) found that even GPT-4 may encounter difficulties when generating vector graphics, requiring additional instructions to correct outputs. This indicates that LLMs still have room for improvement in transforming natural language instructions into renderable, high-quality code that complies with SVG specifications. How to ensure generated geometric precision and path quality through stronger neural path representations and optimization algorithms is one of the core challenges in current research.

5.2 Difficulties in Semantic and Visual Consistency

Large language models excel at understanding text semantics, but accurately translating this semantics into visually consistent and aesthetically pleasing vector graphics is a complex multimodal alignment problem. Generated graphics must not only match the semantic content described in text but also meet mainstream aesthetic standards in visual style, composition, and color matching.

SVGenius benchmark results show that proprietary models also experience performance degradation when handling increasingly complex tasks, with style transfer being one of the most challenging tasks. This means LLMs still have deficiencies in capturing and reproducing specific visual styles. For example, when a user requests a "cartoon-style cat icon," the LLM needs to understand the visual characteristics of "cartoon style" and apply them to the generated SVG.

Chat2SVG combines image diffusion models to enhance visual fidelity and semantic alignment, but ensuring consistent high alignment between LLM-generated semantic templates and diffusion model-generated low-level geometric details remains an open problem. Multimodal hallucination—where generated content semantically mismatches input—is a common issue in MLLMs [34]. In vector graphics generation, this may manifest as generating shapes, colors, or layouts inconsistent with text descriptions, degrading user experience. How to reduce semantic-visual inconsistency through more refined multimodal fusion mechanisms and stricter evaluation metrics is an urgent problem to solve.

5.3 Dataset Scarcity and Evaluation Challenges

High-quality, large-scale vector graphics datasets are crucial for training powerful LLMs, but compared to raster images or pure text data, vector graphics datasets are relatively scarce and costly to construct. Existing datasets such as SVG-Stack [2], SVGX-SFT Dataset [1], MMSVG-2M [29], and UniSVG [56], while providing valuable resources for research, still need improvement in scale and diversity, especially regarding complex scenes, specific styles, or multimodal paired data. Table 3 summarizes currently commonly used open-source datasets in this field and their key characteristics, which form the foundation for training high-performance vector generation models.

Table 3 Statistics of Commonly Used Vector Graphics Datasets

Dataset Name	Source & Scale	Key Features	Application Scenarios	Reference
SVG-Stack	Real-world SVG code	200K+	Pretrained LLM	Rodriguez et al. [2]
MMSVG-2M	Multimodal (text-SVG pairs)	2M	Cross-modal generation and comprehensive understanding	Yang et al. [29]
UniSVG	Icons, charts, etc.	100K+	Comprehensive understanding and generation	Li et al. [56]
VectorEdits	SVG image-instruction pairs	270K+	Instruction-guided editing	Kucha et al. [52]
ColorSVG-100K	Colored vector icons	100K+	Complex icon generation	Chen et al. [43]

Furthermore, evaluating vector graphics is more challenging than evaluating raster images. Raster images can be quantitatively assessed using metrics such as FID and CLIPScore, but these may not fully capture unique vector graphics attributes like editability, structuredness, geometric precision, and semantic consistency. Benchmarks such as SVGenius [24] and VGBench [57] aim to provide more comprehensive evaluation frameworks, but designing automated metrics that comprehensively measure LLM-generated vector graphics quality—including visual aesthetics, geometric accuracy, semantic compliance, editability, and code conciseness—and align them closely with human perception remains an active research direction. SVGauge [55] attempts to evaluate visual fidelity and semantic consistency using SigLIP embeddings and BLIP-2 captions, showing higher correlation with human judgment, providing useful exploration for future evaluation methods.

5.4 Complexity and Controllability Issues

Vector graphics complexity manifests in hierarchical structures, nested relationships, and interactions among various primitives such as paths, shapes, text, and gradients. LLMs often struggle to maintain global consistency and coordinate local details when generating complex scenes or multi-object compositions. For example, generating a scene containing multiple interacting objects requires LLMs to understand spatial relationships, occlusion relationships, and semantic associations among objects, and accurately reflect these in SVG code. Dou et al. (2024) [58] noted that traditional raster graphics recognition methods may suffer from information loss, while hierarchical structure recognition in vector graphics remains challenging.

Controllability is another key issue. Users typically desire fine-grained control

over generated vector graphics, such as adjusting an object' s color, changing a path' s curvature, or adding new elements in specific areas. Although methods like Chat2SVG [28] and SketchAgent [47] support natural language editing, their control granularity and flexibility remain limited. How to achieve precise control at semantic, object, or even path levels while maintaining overall generation consistency is an important challenge for LLMs in vector graphics generation. The learnable semantic token mechanism proposed by LLM4SVG [5] and the HIVE mechanism introduced by SVGDreamer++ [19] both attempt to enhance object-level editability, but extending them to more complex scenes and finer-grained control requires further exploration.

5.5 Computational Efficiency and Resource Consumption

Training and deploying large language models themselves require enormous computational resources and energy consumption [59,60]. When LLMs are combined with image diffusion models or other complex generation models, this computational burden increases further. For example, optimization-based methods typically require multiple iterations to generate high-quality results, resulting in slower generation speeds. Although some works have achieved significant improvements in generation efficiency, computational efficiency remains a bottleneck for real-time interactive design or large-scale content generation scenarios. How to design more lightweight and efficient LLM architectures and training strategies while maintaining generation quality is a future research concern.

5.6 Ethical and Security Issues

Like all generative AI technologies, LLMs in vector graphics generation also face ethical and security challenges. These include copyright attribution of generated content, potential bias and discrimination, and abuse risks. Additionally, provenance and responsibility attribution of generated content are complex issues [61]. How to ensure fairness, transparency, and interpretability of generated content and establish effective governance mechanisms are aspects that cannot be ignored in the development of this field.

In summary, although large language models have broad prospects in vector graphics generation, they still need to overcome challenges in geometric precision, semantic consistency, data scarcity, evaluation difficulties, complexity control, computational efficiency, and ethical security. Solving these problems will require interdisciplinary collaboration, combining the latest advances in computer graphics, natural language processing, machine learning, and human-computer interaction.

To more intuitively present current technical bottlenecks and potential breakthroughs in the field, Table 4 summarizes the core challenges and corresponding solutions.

Table 4 Challenge-Cause-Solution Mapping

Challenge	Cause	Solution & Related Research
Geometric Precision	LLMs excel at semantic logic but lack intuitive perception of continuous coordinate values and topological structures	Neural path representation (NeuralSVG [30]), introducing geometric constraint optimization
Semantic Consistency	Difficulty aligning text semantic space with vector graphics space	Multimodal fusion mechanisms (MAGE [37]), fine-grained alignment algorithms
Controllability	Lack of object-level semantic decoupling, complex SVG code structure	Introducing semantic tokens (LLM4SVG [5]), hierarchical generation mechanisms
Computational Efficiency	Many iterative optimization steps, long autoregressive generation sequences	End-to-end generation architectures, hybrid model optimization (SVGBuilder [43])

6 Future Trends and Research Outlook

Large language models in vector graphics generation are in a rapid development stage. Future research will focus on improving generation quality, enhancing control capabilities, expanding application scenarios, and addressing existing challenges. The following are several key future trends and research outlooks.

6.1 More Powerful Multimodal Understanding and Generation Capabilities

Future LLMs will not just be text generators but truly multimodal intelligent agents capable of deeply understanding and fusing information from multiple modalities including text, images, sketches, audio, and even 3D models. This means LLMs will be able to: (1) **Achieve finer cross-modal alignment:** Current multimodal models (e.g., BLIP-2 [35], Kosmos-2 [36]) have made significant progress in image-text alignment, but for vector graphics as structured, symbolic visual representations, achieving pixel-level, object-level, or even path-level fine alignment remains challenging. Future research will explore more complex cross-modal encoders and decoders to ensure LLMs can accurately capture all user intentions and details in different modality inputs. For example, when users provide a sketch and a text description, LLMs should seamlessly fuse the sketch' s geometric structure with the text' s semantic information to generate highly consistent vector graphics. (2) **Multimodal instruction following and reasoning:** LLMs' strength lies in their instruction-following and reasoning capabilities [62,63]. In the future, LLMs will handle more complex instructions involving multimodal information, such as "Change this icon' s style to watercolor and place it in the top-left corner of the background image." This requires LLMs to not only understand text instructions but also perform visual reasoning on images and transform them into vector graphic operations. Technologies like Visual In-Context Learning (VICL) [64] are expected to further enhance LVLMs' performance in visual reasoning tasks. (3) **3D vector graphics generation:** Current research mainly focuses on 2D vector graphics, but 3D vector graphics have enormous potential in gaming, VR/AR, and industrial design. Works such as Dream3DVG [65] and ViewCraft3D [66] have begun exploring 3D vector graphics generation, achieving arbitrary viewpoint viewing, progressive detail optimization, and viewpoint-dependent occlusion awareness by combining 3D Gaussian splatting and 3D vector graphics branches. Future LLMs are expected to directly generate editable 3D vector models from text descriptions, even supporting multimodal inputs like text + 2D sketches to generate 3D vector graphics, which will greatly expand LLMs' application boundaries in creative design. Works like ShapeGPT [67] have already demonstrated the potential of multimodal language models in 3D shape generation.

6.2 Enhanced Controllability and Editability

The core advantage of vector graphics lies in their editability. Future research will be dedicated to empowering LLMs with more powerful controllability,

enabling users to perform fine-grained, semantic editing of generated vector graphics, rather than just producing one-time results. (1) **Layered and Component-based Generation:** Drawing inspiration from works such as SVGBuilder[43] and LIVE, future LLMs will be able to generate vector graphics in a layered and component-based manner. This means users can independently edit various components of the graphics, or quickly construct complex graphics by combining predefined components. The learnable semantic token mechanism of LLM4SVG[5] and the HIVE mechanism of SVGDreamer++[19] have already made strides in this direction, and future efforts will further enhance object-level editability. (2) **Semantic-Driven Editing Interface:** Editing through natural language instructions will become mainstream. Users can modify graphics through descriptive language as if conversing with a designer, for example, "turn this rectangle into a rounded corner, change the color to blue, and move it slightly to the right." This requires LLMs to have a deeper understanding of SVG code, capable of mapping high-level semantic instructions to precise geometric operations. Chat2SVG[28] and SketchAgent[47] have already demonstrated this potential, and future work will enhance the granularity and robustness of editing. The emergence of the VectorEdits[52] dataset will also accelerate research in this area. (3) **Style Customization and Transfer:** Personalized style is an important component of design. Future LLMs will be able to learn and apply various artistic styles, and allow users to perform style customization and transfer through text descriptions or reference images. The two-stage style customization pipeline proposed by Zhang et al. (2025)[68], utilizing T2V diffusion models and T2I image priors, provides style customization capabilities for SVG generation. VectorPainter[48] also explores reference-guided stylized vector graphics synthesis. This will enable LLMs to generate more artistic and personalized vector graphics.

6.3 Efficient and Robust Generation Architectures

To meet the demands of real-time interaction and large-scale content generation, future LLMs for vector graphics generation will place greater emphasis on efficiency and robustness. (1) **End-to-end generation and optimization:** Reducing intermediate representation conversion and optimization steps to achieve more direct and efficient end-to-end generation. For example, DiffPlanner [44] directly generates vector floor plans, avoiding rasterization conversion. NeuralSVG' s implicit neural representation also provides new ideas for directly generating structured SVG. (2) **Optimization of hybrid model architectures:** Hybrid frameworks like Chat2SVG and SVGFusion have already proven the effectiveness of combining LLMs with image diffusion models. Future research will further optimize these hybrid architectures. To more intuitively present this cross-paradigm fusion evolution direction, Figure 3

shows a unified generation framework for the future. This framework breaks the barriers of single technical paths, flexibly scheduling three generation paths —optimization, diffusion, and autoregression—through a shared multimodal en-

Figure 3

Figure 3: Figure 3

coder, ultimately outputting standard SVG via a differentiable renderer. This "three-stream convergence" architecture design can not only balance generation quality and geometric controllability but also provide a blueprint for exploring more efficient inter-modal information transfer mechanisms. (3) **Reinforcement learning and human feedback:** Introducing reinforcement learning (RL) and reinforcement learning from human feedback (RLHF) mechanisms to continuously optimize LLM performance in vector graphics generation. ReasonSVG [69] has already explored hybrid reward RL to enhance LLM reasoning capabilities in SVG generation. Through human evaluation and feedback, LLMs can learn generation strategies that better align with human aesthetics and design habits, thereby improving the quality and usability of generated results.

Figure 3 Unified Framework for Large Language Model Vector Graphics Generation

6.4 Comprehensive Evaluation Systems and Datasets

The current vector graphics generation field faces challenges of dataset scarcity and imperfect evaluation standards. Future research will 致力于构建更全面、更具挑战性的数据集和评估基准。未来需要更大规模、更高质量、更具多样性的多模态数据集，包含复杂的场景、多种风格、详细的语义标注和分层结构信息，以支持更强大的 LLM 训练。同时未来需要开发更全面、更客观、与人类审美和设计意图高度对齐的自动化评估指标，以准确衡量生成结果的视觉质量、几何精度、语义一致性、可编辑性以及代码质量。SVGenius[24] 和 VGBench[57] 等基准已经开始关注 LLMs 在 SVG 理解和生成方面的表现。SVGauge[55] 提出的以人类感知为导向的评估指标，为未来评估体系的发展提供了方向。

6.5 Human-Computer Collaborative Design and Intelligent Assistance Tools

LLMs will not just be generation tools but intelligent assistants for designers, promoting a new paradigm of human-computer collaborative design. (1) **Intelligent design suggestions and automated optimization:** LLMs can provide intelligent design suggestions based on designers' intentions and current design context, such as recommending appropriate color schemes, layout structures, or style elements. Meanwhile, LLMs can also automate tedious design optimization tasks like path simplification, alignment adjustment, or responsive design adaptation. (2) **Personalized learning and adaptation:** Future LLMs will be able to learn designers' personal styles and preferences and perform personalized generation and editing according to their habits. Through continuous interaction with designers, LLMs can continuously adapt and evolve, becoming truly intelligent partners that understand designers. (3) **Cross-platform integration:** Deep integration of LLMs into mainstream design software to

achieve seamless AI-assisted design experience. The integration of SVGFusion [7] results into Adobe Illustrator 2025 mentioned in the abstract is precisely an embodiment of this trend.

6.6 Research Issues and Key Problems for Human-Computer Collaboration

Large language model participation in vector graphics generation is not the evolution of a single technical path but involves systematic issues of semantic modeling, structural representation, and interaction mechanisms. Although existing research has made progress in different directions, its core research issues have not yet formed a unified framework, mainly concentrated in the following aspects.

First, the collaborative mechanism between semantic planning and geometric generation remains to be systematically modeled. Current mainstream methods mostly adopt staged or modular architectures, where large language models are responsible for high-level semantic parsing and generation planning, while diffusion models or geometric optimization modules complete specific path generation. This loosely coupled structure offers good flexibility in practice but easily leads to inconsistency between semantic intentions and geometric structures in complex scenarios. How to achieve collaborative optimization of semantic decision-making and geometric generation processes while maintaining model interpretability is a key research problem for high-quality SVG generation. Second, SVG modeling as programmatic and structured representation remains immature. Existing methods often simplify SVG into linear code sequences or continuous latent vectors, making it difficult to fully characterize its inherent hierarchical relationships, path dependencies, and editable semantics. This issue directly constrains the practical value of generated results in subsequent editing, reuse, and design iteration. Exploring modeling methods that combine graphical grammar constraints, neural path representations, and structure-aware generation mechanisms around the goal of "editability-first" is an important direction to push the field from visual generation to design support. Finally, the interactive generation paradigm from a human-computer collaboration perspective remains in preliminary exploration. Current research mostly focuses on quality evaluation of one-time generation results, while real design practice emphasizes multi-round interaction, local modification, and style continuity between humans and systems. How to deeply integrate large language models' conversational reasoning capabilities with vector graphics' local operation mechanisms, enabling models to understand design context and support progressive creation, is an important challenge connecting technical research with practical applications.

Overall, these issues indicate that LLM-generated SVG is not only a problem of generation model performance improvement but a comprehensive research direction involving representation methods, generation logic, and interaction paradigms. Its development relies on collaborative advancement of computer

graphics, multimodal learning, and human-computer interaction research.

This paper systematically reviews research progress and technological evolution paths of large language models in vector graphics generation. The survey shows that the field has evolved from early methods relying on vision-language model semantic guidance to structured SVG generation paradigms centered on diffusion models and large language model collaborative modeling, achieving significant improvements in generation quality, semantic consistency, and editability.

From a technical perspective, the research trend of directly modeling SVG structure and generation logic is driving vector graphics generation from "visual result-oriented" to "design process-oriented" transformation. Multimodal large models and neural path representation technologies provide new possibilities for understanding and generating complex vector graphics. Meanwhile, this paper' s discussion reveals that current research still faces key issues including insufficient collaboration between semantic planning and geometric generation, inadequate modeling of SVG structured representation, and immature human-computer interaction mechanisms. Future research needs to pay more attention to editability, interactivity, and design practice requirements while ensuring generation quality, and promote the transformation of large language models from generation tools to design assistance systems through deepening human-computer collaborative design paradigms.

Overall, large language models provide new research perspectives and methodological foundations for vector graphics generation, but their application potential in complex design scenarios still needs to be further unleashed through systematic research.

References

[1] Xia Pingping, Lü Taizhi. Design and Implementation of a Scalable Vector Graphics Editing Component[J]. Journal of Engineering Design, 2012(01): 49-52.
[2] Gan Zaobin, Peng Bin. Data Description Model for SVG-Based Vector Graphics Editing Systems[J/OL]. Computer Engineering and Design, 2005(01): 270-273. DOI:10.16208/j.issn1000-7024.2005.01.087.
[3] MA X, ZHOU Y, XU X, et al. Towards Layer-wise Image Vectorization[C/OL]//2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2022: 16293-16302. http://dx.doi.org/10.1109/CVPR52688.2022.01583. DOI:10.1109/cvpr52688.2022.01583.
[4] ZHANG S, MA J, WU J, et al. Editing Motion Graphics Video via Motion Vectorization and Transformation[J/OL]. ACM Transactions on Graphics, 2023, 42(6): 1-13. http://dx.doi.org/10.1145/3618316. DOI:10.1145/3618316.
[5] XING X, HU J, LIANG G, et al. Empowering LLMs to Understand and Generate Complex Vector Graphics[J]. arXiv preprint arXiv:2412.11102, 2024.
[6] RODRIGUEZ J A, PURI A, AGARWAL S, et al. StarVector: Generating Scalable Vector Graphics Code from Images and Text[Z]. ServiceNow Research,

Mila - Quebec AI Institute, Canada CIFAR AI Chair, ÉTS, Montréal, Canada, UBC, Vancouver, Canada, 2025.
[7] XING X, HU J, ZHANG J, et al. SVGFusion: Scalable Text-to-SVG Generation via Vector Space Diffusion[Z/OL]. arXiv, 2024. https://arxiv.org/abs/2412.10437. DOI:10.48550/ARXIV.2412.10437.
[8] OPENAI, ACHIAM J, ADLER S, et al. GPT-4 Technical Report[Z/OL]. arXiv, 2023. https://arxiv.org/abs/2303.08774. DOI:10.48550/ARXIV.2303.08774.
[9] OPENAI, :, HURST A, et al. GPT-4o System Card[Z/OL]. arXiv, 2024. https://arxiv.org/abs/2410.21276. DOI:10.48550/ARXIV.2410.21276.
[10] Zhao Zhaoyang, Zhu Guibo, Wang Jinqiao. Insights from ChatGPT for Large Language Models and New Development Ideas for Multimodal Large Models[J]. Data Analysis and Knowledge Discovery, 2023(03): 26-35.
[11] FRANS K, SOROS L B, WITKOWSKI O. CLIPDraw: Exploring Text-to-Drawing Synthesis through Language-Image Encoders[Z/OL]. arXiv, 2021. https://arxiv.org/abs/2106.14843. DOI:10.48550/ARXIV.2106.14843.
[12] SCHALDENBRAND P, LIU Z, OH J. StyleCLIPDraw: Coupling Content and Style in Text-to-Drawing Translation[C/OL]//Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence. International Joint Conferences on Artificial Intelligence Organization, 2022: 4966-4972. http://dx.doi.org/10.24963/ijcai.2022/688. DOI:10.24963/ijcai.2022/688.
[13] SCHALDENBRAND P, LIU Z, OH J. StyleCLIPDraw: Coupling Content and Style in Text-to-Drawing Synthesis[J/OL]. 2021. https://arxiv.org/abs/2111.03133. DOI:10.48550/ARXIV.2111.03133.
[14] VINKER Y, PAJOUHESHGAR E, BO J Y, et al. CLIPasso[J/OL]. ACM Transactions on Graphics, 2022, 41(4): 1-11. http://dx.doi.org/10.1145/3528223.3530068. DOI:10.1145/3528223.3530068.
[15] SONG Y, SHAO X, CHEN K, et al. CLIPVG: Text-Guided Image Manipulation Using Differentiable Vector Graphics[Z/OL]. arXiv, 2022. https://arxiv.org/abs/2212.02122. DOI:10.48550/ARXIV.2212.02122.
[16] XING X, WANG C, ZHOU H, et al. DiffSketcher: Text Guided Vector Sketch Synthesis through Latent Diffusion Models[J/OL]. 2023. https://arxiv.org/abs/2306.14685. DOI:10.48550/ARXIV.2306.14685.
[17] JAIN A, XIE A, ABBEEL P. VectorFusion: Text-to-SVG by Abstracting Pixel-Based Diffusion Models[C/OL]//2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2023: 1911-1920. http://dx.doi.org/10.1109/CVPR52729.2023.00190. DOI:10.1109/cvpr52729.2023.00190.
[18] XING X, ZHOU H, WANG C, et al. SVGDreamer: Text Guided SVG Generation with Diffusion Model[Z/OL]//2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2024: 4546-4555. http://dx.doi.org/10.1109/CVPR52733.2024.00435. DOI:10.1109/cvpr52733.2024.00435.
[19] XING X, YU Q, WANG C, et al. SVGDreamer++: Advancing Editability and Diversity in Text-Guided SVG Generation[J/OL]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2025, 47(7): 5397-5413. http://dx.doi.org/10.1109/TPAMI.2025.3547889. DOI:10.1109/tpami.2025.3547889.

[20] ARAR E, FRENKEL Y, COHEN-OR D, et al. SwiftSketch: A Diffusion Model for Image-to-Vector Sketch Generation[J/OL]. 2025. https://arxiv.org/abs/2502.08642. DOI:10.48550/ARXIV.2502.08642.
[21] JARSKY I, KUZIN M, EFIMOVA V, et al. VectorWeaver: Transformers-Based Diffusion Model for Vector Graphics Generation[C/OL]//Proceedings of the 20th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. SCITEPRESS - Science; Technology Publications, 2025: 184-195. http://dx.doi.org/10.5220/0013185100003912. DOI:10.5220/0013185100003912.
[22] TIMOFEENKO B, EFIMOVA V, FILCHENKOV A. Vector Graphics Generation with LLMs: Approaches and Models[J/OL]. Journal of Mathematical Sciences, 2024, 285(2): 169-179. http://dx.doi.org/10.1007/s10958-024-07423-3. DOI:10.1007/s10958-024-07423-3.
[23] ANONYMOUS. Research on LLM-Based Vector Art Generation[J]. Journal of Machine Learning Research, 2024, 25(12): 3214-3225.
[24] CHEN S, DONG X, XU H, et al. SVGenius: Benchmarking LLMs in SVG Understanding, Editing and Generation[C]//ACM Conference. ACM, 2025.
[25] WANG F, ZHAO Z, LIU Y, et al. SVGen: Interpretable Vector Graphics Generation with Large Language Models[Z/OL]//Proceedings of the 33rd ACM International Conference on Multimedia. ACM, 2025: 9608-9617. http://dx.doi.org/10.1145/3746027.3755011. DOI:10.1145/3746027.3755011.
[26] CHEN H, ZHAO Z, CHEN Y, et al. SVGThinker: Instruction-Aligned and Reasoning-Driven Text-to-SVG Generation[C/OL]//Proceedings of the 33rd ACM International Conference on Multimedia. ACM, 2025: 11004-11012. http://dx.doi.org/10.1145/3746027.3755392. DOI:10.1145/3746027.3755392.
[27] CHEN Y, ZHANG H, HUANG Y, et al. Symbolic Graphics Programming with Large Language Models[Z/OL]. arXiv, 2025. https://arxiv.org/abs/2509.05208. DOI:10.48550/ARXIV.2509.05208.
[28] WU R, SU W, LIAO J. Chat2SVG: Vector Graphics Generation with Large Language Models and Image Diffusion Models[J]. arXiv preprint arXiv:2411.16602, 2024.
[29] YANG Y, CHENG W, CHEN S, et al. OmniSVG: A Unified Scalable Vector Graphics Generation Model[Z/OL]. arXiv, 2025. https://arxiv.org/abs/2504.06263. DOI:10.48550/ARXIV.2504.06263.
[30] POLACZEK S, ALALUF Y, RICHARDSON E, et al. NeuralSVG: An Implicit Representation for Text-to-Vector Generation[J/OL]. 2025. https://arxiv.org/abs/2501.03992. DOI:10.48550/ARXIV.2501.03992.
[31] ZHANG P, ZHAO N, LIAO J. Text-to-Vector Generation with Neural Path Representation[J/OL]. ACM Transactions on Graphics, 2024, 43(4): 1-13. http://dx.doi.org/10.1145/3658204. DOI:10.1145/3658204.
[32] WANG Y, LIAN Z. DeepVecFont: synthesizing high-quality vector fonts via dual-modality learning[J/OL]. ACM Transactions on Graphics, 2021, 40(6): 1-15. http://dx.doi.org/10.1145/3478513.3480488. DOI:10.1145/3478513.3480488.
[33] CAO D, WANG Z, ECHEVARRIA J, et al. SVGformer: Representation Learning for Continuous Vector Graphics using Transformers[Z/OL]//2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

IEEE, 2023: 10093-10102. http://dx.doi.org/10.1109/CVPR52729.2023.00973. DOI:10.1109/cvpr52729.2023.00973.
[34] YIN S, FU C, ZHAO S, et al. A survey on multimodal large language models[J/OL]. National Science Review, 2024, 11(12). http://dx.doi.org/10.1093/nsr/nwae403. DOI:10.1093/nsr/nwae403.
[35] LI J, LI D, SAVARESE S, et al. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models[Z/OL]. arXiv, 2023. https://arxiv.org/abs/2301.12597. DOI:10.48550/ARXIV.2301.12597.
[36] PENG Z, WANG W, DONG L, et al. Kosmos-2: Grounding Multimodal Large Language Models to the World[J/OL]. 2023. https://arxiv.org/abs/2306.14824. DOI:10.48550/ARXIV.2306.14824.
[37] E S, YANG Y, WU J, et al. MAGE: Multimodal Alignment and Generation Enhancement via Bridging Visual and Semantic Spaces[Z/OL]. arXiv, 2025. https://arxiv.org/abs/2507.21741. DOI:10.48550/ARXIV.2507.21741.
[38] LIN J, YIN H, PING W, et al. VILA: On Pre-training for Visual Language Models[C/OL]//2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2024: 26679-26689. http://dx.doi.org/10.1109/CVPR52733.2024.02520. DOI:10.1109/cvpr52733.2024.02520.
[39] CAI M, HUANG Z, LI Y, et al. Leveraging Large Language Models for Scalable Vector Graphics-Driven Image Understanding[J/OL]. 2023. https://arxiv.org/abs/2306.06094. DOI:10.48550/ARXIV.2306.06094.
[40] WANG Z, HSU J, WANG X, et al. Visually Descriptive Language Model for Vector Graphics Reasoning[Z/OL]. arXiv, 2024. https://arxiv.org/abs/2404.06479. DOI:10.48550/ARXIV.2404.06479.
[41] KADAM S. Generative AI Integration in Intelligent Graphics Editors[J/OL]. International Scientific Journal of Engineering and Management, 2025, 04(06): 1-9. http://dx.doi.org/10.55041/isjem04158. DOI:10.55041/isjem04158.
[42] WU R, SU W, MA K, et al. IconShop: Text-Guided Vector Icon Synthesis with Autoregressive Transformers[J/OL]. ACM Transactions on Graphics, 2023, 42(6): 1-14. http://dx.doi.org/10.1145/3618364. DOI:10.1145/3618364.
[43] CHEN Z, PAN R. SVGBuilder: Component-Based Colored SVG Generation with Text-Guided Autoregressive Transformers[J/OL]. Proceedings of the AAAI Conference on Artificial Intelligence, 2025, 39(3): 2358-2366. http://dx.doi.org/10.1609/aaai.v39i3.32236. DOI:10.1609/aaai.v39i3.32236.
[44] WANG S, PAJAROLA R. Eliminating Rasterization: Direct Vector Floor Plan Generation With DiffPlanner[J/OL]. IEEE Transactions on Visualization and Computer Graphics, 2025, 31(10): 7906-7922. http://dx.doi.org/10.1109/TVCG.2025.3559682. DOI:10.1109/tvcg.2025.3559682.
[45] CUI Y, GE L W, DING Y, et al. Promises and Pitfalls: Using Large Language Models to Generate Visualization Items[J/OL]. IEEE Transactions on Visualization and Computer Graphics, 2025, 31(1): 1094-1104. http://dx.doi.org/10.1109/TVCG.2024.3456309. DOI:10.1109/tvcg.2024.3456309.
[46] CHEN N, ZHANG Y, XU J, et al. VisEval: A Benchmark for Data Visualization in the Era of Large Language Models[J/OL]. IEEE Transactions on Visualization and Computer Graphics, 2025, 31(1): 1301-1311. http://dx.doi.org/10.1109/TVCG.2024.3456320. DOI:10.1109/tvcg.2024.3456320.

[47] VINKER Y, SHAHAM T R, ZHENG K, et al. SketchAgent: