Journal of Imaging

MDPI

Review

From Fragment to One Piece: A Review on AI-Driven Graphic Design

Xingxing Zou [1,*], Wen Zhang [2] and Nanxuan Zhao [3]

[1] School of Fashion and Textiles, The Hong Kong Polytechnic University, Hong Kong SAR 999077, China
[2] School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, AZ 85281, USA; wenzhang.ccm@gmail.com
[3] Adobe Inc., San Jose, CA 95110-2704, USA; nanxuanzhao@gmail.com
* Correspondence: xingxing.zou@polyu.edu.hk

Abstract

This survey offers a comprehensive overview of advancements in Artificial Intelligence in Graphic Design (AIGD), with a focus on the integration of AI techniques to enhance design interpretation and creative processes. The field is categorized into two primary directions: perception tasks, which involve understanding and analyzing design elements, and generation tasks, which focus on creating new design elements and layouts. The methodology emphasizes the exploration of various subtasks including the perception and generation of visual elements, aesthetic and semantic understanding, and layout analysis and generation. The survey also highlights the role of large language models and multimodal approaches in bridging the gap between localized visual features and global design intent. Despite significant progress, challenges persist in understanding human intent, ensuring interpretability, and maintaining control over multilayered compositions. This survey aims to serve as a guide for researchers, detailing the current state of AIGD and outlining potential future directions.

Keywords: AI in graphic design; design interpretation; creative process

Academic Editor: Norimichi Tsumura

Received: 19 July 2025
Revised: 15 August 2025
Accepted: 19 August 2025
Published: 25 August 2025

Citation: Zou, X.; Zhang, W.; Zhao, N. From Fragment to One Piece: A Review on AI-Driven Graphic Design. *J. Imaging* **2025**, *11*, 289. https://doi.org/10.3390/jimaging11090289

1. Introduction

Forecasts by McKinsey and PricewaterhouseCoopers suggest that generative AI in graphic design could potentially contribute more than USD 8 trillion to the global economy by 2030. Research interest in this area has continuously increased.

The academic evolution of Artificial Intelligence in Graphic Design (AIGD) reveals two distinct phases. Early efforts focused on decomposing design tasks into atomic components—typography generation [1], layout optimization [2,3], and color palette recommendation [4–8]—employing specialized models for each subtask. While effective for generating individual elements, this decompositional approach introduced systemic fragmentation that persists in current research [9]. Recent breakthroughs in large-scale text models have catalyzed the evolution of generative visual models [10,11]. Within graphic design [12], as illustrated in Figure 1, this progress reflects a paradigm shift: from optimizing isolated elements (e.g., typography, images, vector shapes, layouts, and colors) to holistic creative systems capable of maintaining aesthetic consistency across entire design workflows—from human instruction to final artwork. Figure 1 serves as a visual representation of this paradigm shift, highlighting the core methodology and nature of the results achieved by contemporary AI-driven design systems. The figure is divided into three

key stages: (1) Human Instruction and Intent Definition: The initial phase captures how users provide input, including textual descriptions, style preferences, and design objectives. This step defines the creative intent and sets the foundation for the AI system to interpret and act upon. (2) AI-Driven Component Optimization: The middle section of the figure illustrates how the system processes the input to generate and refine individual design elements. For example, the AI applies advanced algorithms to create typography, shapes, and layouts that align with the specified instructions. Tools like Adobe Firefly excel in this stage, leveraging machine learning to produce assets that are both visually appealing and contextually relevant. (3) Holistic Workflow Integration: The final stage presents the system's ability to harmonize all design elements into a cohesive output. This includes maintaining aesthetic consistency across colors, proportions, and layouts, ensuring the final artwork adheres to the creative intent defined in the first stage. By following this structured workflow, as depicted in Figure 1, these systems demonstrate how they bridge the gap between isolated design tasks and fully integrated creative processes. This figure not only illustrates the methodology but also underscores the transformative nature of these AI-driven solutions, which enable designers to achieve faster, more efficient, and aesthetically consistent results. Tools like Adobe Firefly are emblematic of this trend, showcasing the potential of AI to redefine creative workflows.

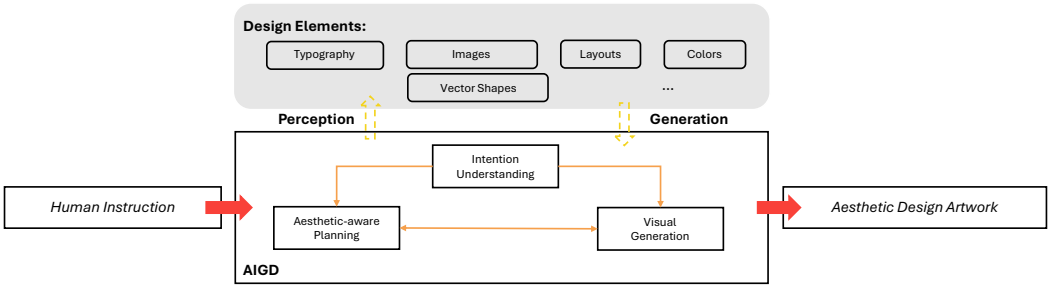

Figure 1. General pipeline of Artificial Intelligence in Graphic Design (AIGD).

As summarized in Table 1, recent surveys in graphic design have explored various dimensions of the field. Ref. [13] analyzes vector graphics through mathematical foundations and content creation stages; ref. [14] delves into layout generation aesthetics and technologies; ref. [9] provides a taxonomy of graphic design intelligence. Ref. [15] reviews graphic layout generation focusing on implementation and interactivity, while [16] investigates challenges and future functional needs for AI-generated image tools in graphic design through designer interviews. Table 1 below highlights the key contributions of these works in comparison to our survey, focusing on the unification of cognitive and generative tasks.

Scope of the Reviews. The multimodal era has seen numerous attempts at cross-modal integration through vision–language models [17,18]. While these efforts represent significant progress, they have largely struggled to bridge the semantic gap between localized visual features and global design intent [19]. Recent strides in LLM-driven design systems [19,20] mark a pivotal shift by introducing a convergence path where generative processes are guided by explicit design rationale encoded in latent spaces [21,22]. Building on these advances, this article introduces a novel perspective on AI-Driven Graphic Design (AIGD), emphasizing design understanding and creativity as central themes. Unlike prior studies that narrowly focused on isolated technical improvements, our work takes a holistic approach to the design process. Specifically, we bridge the gap between abstract concepts and tangible creations, offering a comprehensive framework that enables readers to grasp the full scope of AIGD. Moreover, we demonstrate how the latest AI advancements can not only streamline workflows but also significantly enhance the creative potential of

graphic design. By reframing AIGD in this way, we provide a fresh lens that highlights the transformative potential of AI in fostering creativity across the design spectrum.

Table 1. Comparison of surveys in graphic design intelligence with quantitative metrics.

Surveys	Focus Area	Key Contribution	Papers Reviewed
Tian et al. (2022) [13]	Mathematical foundations and content creation stages of vector graphics	Provides a foundation for understanding vector graphic representation and its mathematical principles	147
Shi et al. (2023) [14]	Layout generation aesthetics and technologies	Explores technologies for automatic layout generation, focusing on aesthetic rules	131
Huang et al. (2023) [9]	Taxonomy of graphic design intelligence	Categorizes approaches to graphic design tasks into a taxonomy of AI-driven intelligence	77
Liu et al. (2024) [15]	Graphic layout generation: implementation and interactivity	Reviews techniques for interactive and automated layout generation	40
Tang et al. (2024) [16]	Challenges and future needs for AI tools in graphic design	Identifies challenges and future functional needs based on designer interviews	39
Ours	Unification of cognitive and generative tasks in design workflows	Proposes a framework that integrates cognitive (e.g., reasoning, decision-making) and generative (e.g., image and layout generation) tasks, highlighting the interplay and potential synergies of AI-driven tools in holistic design processes	267

To address the evolving landscape of AIGD, this survey conducts an extensive and in-depth analysis. By examining AIGD research through the dual lenses of design semantics—such as visual hierarchy, typography, and color theory—and creative workflows—including ideation, iteration, and refinement—we establish a unified framework to assess advancements in this domain. This approach contrasts with previous analyses, which were often fragmented, by providing a structured examination of how AI models both interpret and generate meaningful design artifacts, such as raster and vector graphics, while maintaining artistic intent and adaptability. We conducted a systematic search across major academic databases, including IEEE Xplore, ACM Digital Library, Scopus, Web of Science, using keywords such as "graphic design", "AI in design", "layout generation", and "generative design". From the initial pool of papers from 2000 to 2025, we applied inclusion and exclusion criteria based on relevance to AI-driven graphic design, leading to a final dataset of 500 papers. Papers were categorized into key themes based on their primary focus, including (i) cognitive tasks (e.g., reasoning, decision-making), (ii) generative tasks (e.g., image and layout generation), and (iii) hybrid approaches integrating both. This categorization was performed by two independent researchers to minimize bias, with disagreements resolved through discussion. Within these categories, we delved into four key subtasks associated with design elements: non-text objects, text characters, aesthetic elements, and layout. The statistical distribution between cognitive and generative research is visually summarized in Figure 2. Our findings highlight several trends:

(1) Recent research predominantly focuses on individual subtasks, with studies considering graphic design as a holistic endeavor emerging prominently since 2023. (2) While raster images are generally more common in AI research, vector images are particularly significant in AIGD due to attributes like lossless scaling. (3) Tasks such as font and layout generation are more frequently addressed for raster images, while vector-based studies tend to focus on non-text objects. This discrepancy is often due to the broader research

on raster image generation not specifically targeting graphic design, resulting in domain incompatibility. (4) There is a marked increase in enthusiasm for AIGD, with significant growth in interest for generation tasks since 2010, especially noted from 2022 onwards.

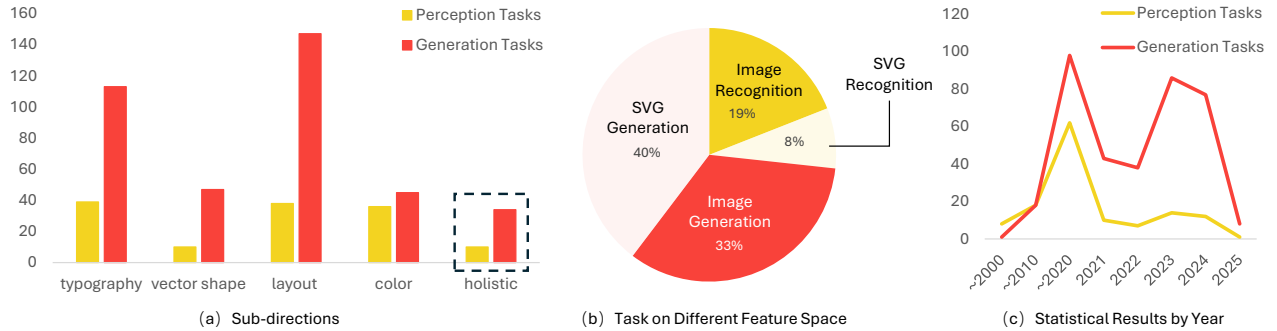

Figure 2. Overview of research publications in AIGD: (**a**) subdirections, (**b**) task on different feature space, and (**c**) statistical results by years. Holistic indicates systems capable of processing multiple or all design elements.

This survey aims to provide a comprehensive review of the methodologies involved in these subtasks, discuss ongoing challenges, unresolved issues, and suggest directions for future research. The remainder of this article is structured as follows: Section 2 presents the background of graphic design, introducing relevant concepts and involved subtasks. Sections 3 and 4 discuss cognitive and generative tasks, respectively. We focus on research strictly related to graphic design or closely associated domains. Given the extensive scope of this survey, we offer condensed information to enhance understanding and outline research branches. Finally, in Section 5, we discuss the state of the art in the era of multimodal large language models, existing challenges, and potential future trends, considering graphic design from a holistic rather than a piecemeal approach.

2. Background

Graphic design aims to deliver information clearly while presenting it in an appealing visual way [23]. It involves design elements, including non-text objects (images and vector shapes) and text characters (typography), to create aesthetic narratives [23] through visual harmony layout and aesthetic elements, particularly colors. Graphic design typically employs two principal data types: raster and vector images.

- A raster image is a two-dimensional array storing pixel values, with the pixel as its fundamental unit influenced by resolution.
- Scalable Vector Graphics (SVG) uses mathematical descriptions to record content, such as parameters to draw straight lines.

We first define the problem of AIGD under a unified mathematical formulation. Let $E = T \cup O$ represent the set of all design elements, where T is the set of text elements (typography), and O is the set of non-text elements (images or vector shapes). Each element $e \in E$ in the design is described by two kinds of features: (1) its attribute vector a_e, which may include style, size, color, etc., and (2) its design content c_e, such as visual context of objects. Therefore, a candidate design artifact is a set of element features, $D(A = \cup a_e, C = \cup c_e)$. The objective of the design is to find an optimal $D(A, C)$ which maximize the following function:

$$D(A, C) = \arg\max_{A,C} V(L(A), C \mid I) \quad (1)$$

where $V(.)$ measures the aesthetic value of design D under the constraint of user intention I. $L(A)$ is the layout of design elements,

$$L(A) = h(\tilde{A}|C) \geq \tau \tag{2}$$

which follows the certain design principles ($h(.)$) providing best harmonic settings among design objects. τ is the minimal harmonic score. In Equation (2), not all attributes are applicable to the layout. Typically, a subset $\tilde{A} = (p_i, s_i, \theta_i)$ represents the core parameters, where $p_i \in \mathbb{R}^2$ represents the position coordinates of element e_i (for a two-dimensional layout), $s_i \in \mathbb{R}^+$ represents the size of an element e_i (e.g., width and height), and $\theta_i \in S$ represents other states of element e_i (e.g., rotation angle, transpose, etc.). Considering the human centric nature of graphic design, human intention I steers the exploration of $D(A, C)$. However, interpretation of human intention is complex and requires consideration of individual preference. It is provided as a part of system inputs, in terms of multimodal instructions, reference images, or templates.

Ideally, it is anticipated that Equation (1) could be resolved within a unified pipeline. The major components would be capable of comprehending design intentions I, acquiring basic relevant elements E_{basic}, orchestrating graphic layouts $L(A)$, and ensuring the visual harmony of the produced outcomes $V(D)$. Recent studies published between 2023 and 2024 have explored the potential of LLMs in graphic design, as evidenced by works such as those by Dou et al. (2024) [24], Huang et al. (2024) [25], and others [19,20,26,27]. However, these efforts are still in the early stages and have yet to achieve a deep understanding and professional generation.

The primary goal of graphic design is to deliver information. While visual elements, particularly images and vector shapes, enhance aesthetic appeal, text elements directly communicate the design's theme and play a more central role in information delivery. This relative importance is reflected in statistical data from the field, indicating that 31% of tasks are text-related, while only 12% focus on non-text elements, including tasks that involve both types of content. Text in graphic design encompasses various attributes such as fonts, glyphs, artistic styles, and semantic typography, all crucial for effective communication. Another significant factor is the meticulous arrangement of these elements. Key messages in the text should be highlighted, and design considerations like adequate white space, optimal contrast, and visual balance are essential for a well-managed layout. Research specifically targeting these considerations accounts for more than 38% of studies in this area. Additionally, achieving aesthetic appeal through techniques like color harmony is also critical, representing 13% of research interests. Due to the complex nature of these tasks, much of the existing research has been divided into distinct sub-directions, primarily categorized into two areas: perception tasks [28–35] and generation tasks [36–47].

3. Perception Tasks

Understanding design intent is the first step towards AIGD, which requires a model with a basic knowledge of graphic design principles. Therefore, this section presents the methodology evolution separately for each subtask, including non-text element perception in Section 3.1, text element perception in Section 3.2, layout analysis in Section 3.3, and aesthetic understanding in Section 3.4. Two main types of data are primarily used in graphic design.

3.1. Non-Text Element Perception

3.1.1. Object Recognition in Raster Image

Numerous comprehensive surveys and reviews have documented advances in non-text object recognition [48]. Building upon the areas not extensively covered by these

surveys, recent progress in multimodal large language models (MLLMs) [49] has significantly expanded the capabilities of LLMs to process and interpret text and visual data. These MLLMs have demonstrated remarkable proficiency in vision–language tasks, such as image captioning and visual question answering. Furthermore, contemporary research has begun to explore the potential of using textual output from LLMs to steer external vision expert models to perform a variety of vision-centric tasks [50]. In object detection, such expert models include systems such as DETR [51], which are designed to improve the accuracy and efficiency of detecting and interpreting visual objects.

3.1.2. SVG Recognition

Traditional methods employ rule-based graph-matching techniques, such as visibility graphs [52] and attributed relational graphs [53]. YOLaT [54] first proposed a learning-based method that represented vector graphics using graphs based on the Bézier curve, where object detection was conducted based on the predictions of a Graph Neural Network (GNN). However, this work modeled only in a flat GNN architecture with vertices as nodes, ignoring the higher-level information of vector data. The follow-up work by [24], YOLaT++, learns multi-level abstraction features from primitive shapes to curves and points. They also provide a new dataset for chart-based vector graphics detection and chart understanding, which includes vector graphics, raster graphics, annotations, and raw data for creating these vector charts.

3.2. Text Element Perception

3.2.1. Optical Character Recognition (OCR)

To facilitate text recognition, it is essential to locate the text area within an image [55]. Popular text detection algorithms can be broadly categorized into regression-based, segmentation-based, and detection transformers. Regression-based algorithms draw from general object detection methods, which treat text detection as a unique scenario within target detection, such as TextBoxes [56] (based on the Single Shot Multi-box Detector (SSD) [57]) and CTPN [58] (based on Faster R-CNN [59]), among others. ABCNet [60] is the first to introduce Bézier curve control points for arbitrary-shaped texts. On the other hand, segmentation-based algorithms, inspired by Mask R-CNN [61], have significantly improved text detection across various scenes and shapes but entail complex post-processing, speed issues, and challenges in detecting overlapping text. DETR [51] represents a pioneering model introducing a fully end-to-end transformer-based paradigm. However, DETR's training convergence and feature resolution limitations have hindered its competitiveness compared to traditional detectors. Other variants include Conditional-DETR [62] and Anchor-DETR [63]. Furthermore, approaches like DN-DETR [64] and MaskDINO [65] concentrate on label assignment strategies, significantly improving matching stability.

Once text is detected, text recognition algorithms identify the content within the detected areas. It is typically divided into regular and irregular text recognition based on the shape of the text. Regular text includes printed fonts and scanned text, whereas irregular text often appears non-horizontal and may exhibit bending, occlusion, and blurring. Historically, the mainstream approach involved segmentation and single-unit recognition, utilizing connected domain analysis to identify potential text segmentation points. Post-2016, the focus shifted to text line recognition. Early works using DNNs as feature extractors for scene text recognition include [66]. However, many texts in natural scenes have arbitrary shapes and layouts, making it difficult to transform them into horizontal texts through the proposed interpolation methods. To this end, the later studies focus on identifying granular-level elements, such as characters, and semantically encoding their relationships to enhance the recognition of irregular text [67–69]. Many recent works have

introduced a growing trend of generative models into scene text recognition. Ref. [70] proposed transforming the entire scene text image into corresponding horizontally written canonical glyphs to promote feature learning. Through their experiments, the guidance of canonical glyph forms proved effective for feature learning in STR.

3.2.2. Font Recognition

Fonts are typically designed with unique characteristics, such as stroke width, serifs, aspect ratio, spacing or slant/italicization [71]. Early font recognition works attempted to recognize fonts via these artificial font features [72,73]. While artificial font features worked reasonably well in controlled scenarios, they faced several challenges: (1.) Font variability: Fonts with subtle differences in design could be difficult to differentiate using simple features. (2.) Noise in input data: Scanned documents or degraded images introduced noise that could distort features like stroke width or spacing. (3.) Handwritten vs. printed fonts: Artificial features were less effective for recognizing handwritten or highly stylized fonts. (4.) Limited scalability: Adding new fonts required manually defining additional rules or features. As the learning-based method became popular, Wang et al. built a Convolutional Neural Network with domain adaptation techniques for font recognition, applying deep neural networks to font recognition for the first time [74]. This method was followed by that of Bharath et al., who utilized SVM for English font recognition, focusing on character image distances [75]. The research was further expanded by Liu et al., who introduced a multi-task adversarial network for Japanese fonts, employing a GAN to preprocess scene text images prior to recognition [76]. The introduction of the FontCLIP latent space further expands the possibilities for font selection using out-of-domain attributes and scripts, improving flexibility [77].

3.3. Layout Analysis

Layout is composed of visual elements, typically characterized by properties such as type and position. As shown in Figure 3, traditional approaches use hand-crafted features to represent layout. For instance, Stoffel et al. designed features related to position, spacing, and font styles for document structure analysis [78]. Some methods employ neural networks, such as transformer [79] and Faster-RCNN [59], to encode layouts into low-dimensional continuous representations, showing promising results [80].

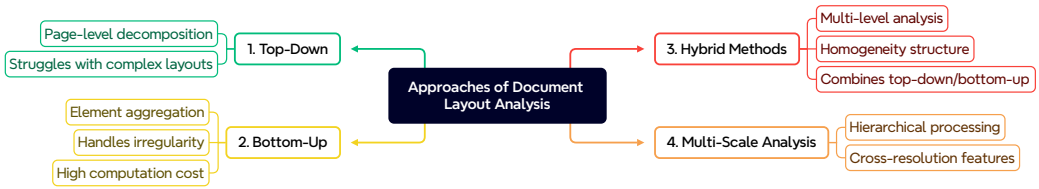

Figure 3. Overview of methods for layout analysis task.

The layout analysis of graphic design shares the common foundation of other aesthetic-aware layout analyses, e.g., Document Layout Analysis (DLA), where domain knowledge can be easily transferred to all general graphic design. DLA methodologies include bottom-up, top-down, hybrid, and multi-scale approaches. The top-down approach starts with each page as a single large block, which is then subdivided into smaller sections, but struggles with complex layouts [81]. The bottom-up approach, starting at the granular level and aggregating adjacent elements into larger blocks, handles irregular layouts well but can be computationally demanding [82,83]. Hybrid methods combine these approaches and utilize a multi-level, homogeneity structure to improve layout analysis [84]. The adoption of CNNs has shifted DLA towards models that extract features directly from document

pixels, addressing shortcomings of traditional methods. Early CNN models focused on textural features for segmenting segments but were less effective with elements like tables [85]. Recent developments like Gruning's ARU-Net and Xu et al.'s multi-task FCN improve text line segmentation and contour detection [86,87]. These innovations emphasize the integration of semantic interpretation in DLA, highlighting the importance of understanding semantic relationships between document components. Ref. [88] highlights the importance of contextual relevance in element placement.

3.4. Aesthetic Understanding

The field of aesthetic understanding in graphic design has evolved from manual feature engineering to AI-driven multimodal systems [89]. Early approaches relied on handcrafted color metrics and rule-based harmony models, while modern methods leverage deep learning (GANs, VAEs, transformers) for context-aware palette generation and personalized recommendations. Concurrently, aesthetic assessment has transitioned from spatial/statistical analysis to neural architectures that model emotional impact and user preferences. This paradigm shift enables unified systems addressing both functional requirements (color discrimination) and affective dimensions (emotional resonance) across infographics, marketing materials, and interactive interfaces.

3.4.1. Color Palettes Recommendation

Initial color recommendation systems relied heavily on manual feature extraction and regression models. For instance, Color Sommelier [90] introduced a harmony rating algorithm based on community-generated palettes, allowing users to iteratively select harmonious color schemes. However, these methods often overlooked the semantic meanings of colors and incorporated less critical features, leading to suboptimal predictions. The advent of deep learning marked a significant paradigm shift in color recommendation systems. Neural networks began to learn color feature representations from image color histograms, classifying images according to predefined categories. Early efforts included the use of neural networks on predefined color palettes tailored for specific themes, such as magazine cover design [91–94].

Recent studies have focused on recommending color palettes for information visualizations and statistical graphics, such as scatterplots and bar charts [95,96]. Beyond simple infographics, researchers have explored color palette recommendations for more complex visual designs, such as advertising posters and magazine covers. Yuan et al. [96] implemented a Variational AutoEncoder with Arbitrary Conditioning (VAEAC) to dynamically suggest colors for various infographic elements. The latest research in color recommendation increasingly focuses on generative models and region-specific recommendations. Refs. [97,98] developed a transformer-based masked color model for specific regions on landing pages and vector graphic documents. Ref. [99] utilized maximum likelihood estimation and conditional variational autoencoders within a transformer framework to recommend text and background colors for e-commerce mobile web pages.

3.4.2. Other Aesthetic Attributes

Aesthetic visual quality assessment advances from traditional handcrafted feature extraction methods to sophisticated deep-learning approaches, with the trend from standardized assessment to more diverse and personalized evaluation. Early aesthetic visual quality assessment methods focused on extracting handcrafted features from images [100,101]. The introduction of models that incorporated human-describable attributes marked a significant advancement. Ref. [102] introduced a model that connected technical analysis with human perceptions, including elements such as composition, illumination, and content. Obrador et al. [103] evaluated photographs based on features like simplicity and visual bal-

ance. Ref. [104] developed models to predict users' first aesthetic impressions of websites, based on visual complexity and colorfulness. With the advent of deep learning, the field underwent a transformative change. Lu et al. [105,106] utilized dual-column CNNs and a Deep Multi-Patch Aggregation Network (DMA-Net) to encode global image layouts better, significantly advancing the classification and understanding of aesthetic qualities. Recent studies have focused on personalized image aesthetics, exploring how users' social behaviors and personal perceptions influence their aesthetic judgments. Cui et al. [107,108] addressed user-centric aesthetic assessment analysis. Chen et al. [109] introduced the Adaptive Fractional Dilated Convolution to maintain the original aspect ratios and composition of images.

3.5. Summary

The evolution of the visual cognition framework is illustrated by the transition from traditional methods to the adoption of deep learning techniques such as CNNs, subsequently incorporating GANs, transformers, and currently, LLMs. Similarly, although each subtask within AIGD progresses independently, their overall development trajectories align consistently with this. Another notable trend is research on vector images, which remains relatively sparse compared to raster images. Most studies on vector cognition have focused on SVG recognition, representing earlier efforts in the field. However, recent statistics indicate a growing interest in vector image research. This surge is attributed to the nature of vector representations, which are highly conducive to integration with LLMs for enhanced understanding and reasoning.

In text recognition, the main challenges addressed are OCR and font recognition. General approaches include techniques such as Faster R-CNN, text line or single character segmentation, and the input ranges from handwritten notes to natural scene images. The recognition process faces several challenges, including text distortion due to perspective changes, small text scale, stylized fonts, various font sizes and styles, decorative elements, multilingual text, image blur, and poor lighting conditions. Font recognition is another crucial aspect of text-related cognition and plays an essential role in graphic design where font styles are vital. However, the diversity of font styles poses a significant challenge to creating a comprehensive dataset, including unique styles such as italics and bold. This makes it difficult for models to learn to recognize diverse fonts.

In addition to text elements, layout analysis, especially in document structure, has received increasing attention from researchers. This analysis is a precursor to OCR, classifying and recognizing different elements in a document, such as text, images, tables, and titles. Recent research has made significant progress through large-scale language model-based tools such as LayoutLM, UDOP, and LiLT, which leverage multimodal transformer encoders pre-trained and fine-tuned for specific applications. Finally, aesthetic research has primarily focused on color matching, with additional analysis based on personality, photographic content, and direct visual feature computation. The subjective nature of aesthetics and the lack of clear principles or standards make it a challenging research area that lacks a strong framework or benchmarking system.

4. Generation Tasks

Graphic design needs elements with separate transparent backgrounds. Thus, we focus primarily on vector shape generation and the vectorization of artistic imagery. For ease of discussion, we categorize text element generation into the generation of text itself and the rendering of text within a scene. Meanwhile, we introduce works in layout generation and layout-based image generation. Finally, we address research focused on aesthetic refinement.

4.1. Non-Text Element Generation

4.1.1. SVG Generation

SVG can be encoded as the sequence of 2D points connected by parametric curves, making the seq2seq model straightforward as an encoder/decoder basis [110–114]. SketchRNN [111] was a pioneer in employing LSTM-based VAEs for learning to draw strokes, representing sketches as sequences of pen positions and states. SVG-VAE [112] involves a two-stage training process that begins with an image-based VAE, followed by training a decoder to predict vector parameters from the latent variables. BézierSketch [115] focuses on generating Bézier curves, offering enhanced control over graphical forms of sketches. DeepSVG [110], a hierarchical autoencoder designed to learn representations of vector paths, contributes to the structural complexity of vector graphics. These methods heavily rely on datasets in vector form, which limits their generalization capabilities and their capacity to synthesize complex vector graphics. IconShop trains a BERT model for text-conditioned SVG generation of icons but is restricted to using paths [114].

Instead of directly learning an SVG, another method of vector synthesis optimizes towards a paired raster image during training. Ref. [116] observed vector graphics rasterization was differentiable after pixel prefiltering. Conditioned on this finding, the authors introduced a differentiable rasterizer that offered two prefiltering options: an analytical prefiltering technique and a multisampling anti-aliasing technique. The analytical variant was faster but could suffer from artefacts such as conflation. The multisampling variant was still efficient and could render high-quality images while computing unbiased gradients for each pixel with respect to curve parameters. That work enabled the supervision of the SVG generation under the guidance of a raster image. In other words, different from image generation methods that traditionally operate over vector graphics and require a vector-based dataset, recent work has demonstrated the use of differentiable renderers to overcome this limitation [117–122]. CanvasVAE defines vector graphic documents through a multimodal set of attributes, using variational autoencoders to integrate diverse graphical components [123]. Im2Vec is a method that employs a differentiable rasterization pipeline to generate complex vector graphics from raster training images [118]. Furthermore, recent advances in visual text embedding and contrastive language–image pre-training models have enabled a number of successful methods for synthesizing sketches, including CLIPDraw and CLIPasso [124,125]. In addition to using CLIP distance, VectorFusion [126] and DiffSketcher [122] combine a differentiable renderer with a text-to-image diffusion model for vector graphics generation. This type of method utilizes Score Distillation Sampling loss based on a text-to-image (T2I) diffusion model for optimizing SVG to align with text prompts across various applications such as fonts, vector graphics, and sketches [127–129]. However, due to the lack of geometric constraints, they often lead to path intersections or jagged effects. By adding geometric constraints to a Text-to-Vector (T2V) generation pipeline that optimizes local neural path representation, high-quality SVG graphics generation is achieved [130].

4.1.2. Vectorization of Artist-Generated Imagery

Image vectorization is another alternative way to directly obtain the bitmap from imagery. Traditional vectorization techniques primarily depend on segmentation or edge detection to group pixels into larger regions, subsequently fitting vector curves and region primitives to these segments [131,132]. Challenges include aligning patch boundaries and automating mesh generation [132,133]. In contour-based vectorization, simpler elements such as lines, circles, and Bézier curves represent discontinuity sets in piecewise constant images, often including silhouettes and pixel art [134,135]. To better fit piecewise smooth vector curves to raster boundaries, ref. [136] proposes an image vectorization method

based on mathematical algorithms for frame field processing. PolyFit [137] approximates piecewise smooth vector curves to raster boundaries with coarse polygons, considering perceptual cues and simplicity. LIVE [117] and SAMVG [138] employ a layer-wise optimization framework that significantly improves vectorization quality. Chen et al. [139] explore the assembly of simple parametric primitives within a neural network for geometric abstraction. SuperSVG [140] focuses on decomposing the input into superpixels for optimized reconstruction and detail refinement.

4.2. Text Element Generation

4.2.1. Artistic Typography Generation

One of the main directions in text element generation is font style learning. Traditional methodologies centered on explicit shape modeling and statistical learning techniques to craft font glyphs of calligraphy, predominantly targeting elements such as strokes and radicals [141,142]. Research efforts have focused on emulating traditional calligraphic styles using hierarchical models and texture transfer techniques [143–146]. Deep learning has markedly increased the flexibility and realism in font creation; researchers have utilized image translation methods for font generation and explored font style learning in one-shot and few-shot settings. Ref. [147] was the first to adopt GANs to automatically generate a Chinese font library by learning a mapping from one style font to another, and DC-Font [148] also addresses the font feature reconstruction and handwriting synthesis problems through adversarial training. However, these methods operate under supervised learning and necessitate a large volume of paired data. Some methods employ auxiliary annotations (e.g., stroke and radical decomposition) to enhance generation quality further. RDGAN [149] proposes a radical extraction module to extract rough radicals. To facilitate the automatic synthesis of new fonts more easily, some works follow unsupervised methods to separately obtain content and style features and then fuse them in a generator to produce new characters [150–152]. Concurrently, other works leverage auxiliary annotations to make the models cognizant of the specific structure and details of glyphs [36,38]. Most recently, Diff-Font [37] represents a pioneering effort to use a diffusion model, treating it as a conditional generation task to manage content through predefined embedding tokens while extracting the desired style from a one-shot reference image.

Another significant line is transferring artistic styles of color and texture onto new glyphs. Yang et al. [153] pioneered text effect transfer by enabling the migration of effects from a stylized text image to a plain one. Following this, ref. [154] developed a general framework for user-guided texture transfer, applicable to a variety of tasks, including transforming doodles into artworks, editing decorative patterns, generating texts with special effects, controlling effect distribution in-text images, and swapping textures. The following research in this domain revolves around exploring and enhancing techniques for separating, transforming, and recombining text styles and content. Research has progressively evolved to address various aspects of style and content encoding, mixing, and decoding. Ref. [155] pioneered the application of deep networks for text effect transfer, focusing on combining font and text effects. Ref. [156] addressed specific issues like stroke adhesion and text clarity, while [157] tackled data scarcity through synthetic data generation. The field has recently seen innovative approaches leveraging diffusion models for diverse style support and interactive generation, culminating in Wang et al. [158]'s method for generating artistic fonts using a text-to-image diffusion model.

In semantic typography, the primary goal is to enable the integration of artistic expression with legibility while embedding semantic meanings into typographic designs. Xu et al. [159] pioneered an interactive method for creating calligrams that warped letters to fit within specific regions of an image, aligning them semantically with the visual content,

albeit sometimes at the cost of readability. Building on the need for clarity, Zou et al. [160] refined guidelines for glyph deformation through a crowd-sourced study, aiming to improve the readability of automated letter layouts. To further personalize and enrich typography, Zhang et al. [161] introduced a framework that allowed users to influence glyph structure interactively, incorporating a semantic-shape similarity metric and optional structural optimization techniques to enhance both aesthetics and integrity. Advancements continued with Iluz et al. [127] who modified letter geometry based on semantic meanings and employed advanced rendering techniques to ensure high-quality visualization across sizes. Finally, Tanveer et al. [162] leveraged large language models and unsupervised generative models to synthesize stylized fonts with embedded semantic meanings.

4.2.2. Visual Text Rendering

Text rendering aims to render the text characters in imagery. Methodologies in this area have been extensively researched to address the issue of visual inconsistency often observable when text is merely superimposed onto images. Notable approaches include SynthText [163], VISD [164], SynthText3D [165]. Despite these technological advancements, the field continues to face significant challenges in achieving accurate text rendering and ensuring visual coherence with the surrounding environment, primarily due to the limited diversity in background datasets utilized for training and synthesis. Most existing research efforts [166] have concentrated on the precise visual rendering of text in English. However, initiatives like AnyText [167] show only moderate success in rendering texts in other languages such as Chinese, Japanese, and Korean. This is largely attributed to the challenges in gathering high-quality data and the constraints of training models on a limited dataset comprising merely 10,000 images across five languages. Given the extensive array of characters in these non-English languages, such a dataset size proves inadequate for comprehensively addressing the task of multilingual visual text rendering. Furthermore, contemporary commercial image generation models like DALL·E3, Imagen3, Stable Diffusion 3 [168], and Ideogram 1.01 have demonstrated underwhelming performance in multilingual text rendering tasks.

Recent research has focused on enhancing text rendering accuracy by integrating large-scale language models such as T5, used by platforms like Imagen [169]. Studies suggest that character-aware models like ByT5 [170] offer substantial advantages over character-blind models such as T5 and CLIP [171] in terms of text rendering accuracy. Innovations such as GlyphDraw [172] introduce frameworks for precise control over character generation, incorporating features like auxiliary text locations and glyph characteristics. TextDiffuser [166] uses a layout transformer to enhance knowledge of text arrangement and integrates character-level segmentation masks for higher accuracy. GlyphControl [173] and Diff-Text [174] refine the approach by facilitating explicit learning of text glyph features and using rendered sketch images as priors for multilingual generation, respectively. Meanwhile, GlyphOnly [175], which uses glyphs and backgrounds for accurate rendering and consistency control, is equipped with an adaptive strategy for exploring text blocks in small-scale text rendering.

4.3. Layout Generation

4.3.1. Automatic Layout Generation

Layout can be created by selecting a template that best fits the content [176–178]. However, such a predefined, constrained set of templates can rarely accommodate the vast diversity of graphic design layouts. Many works have studied the creation of a layout according to the given elements for graphic design such as UI design [179], advertisement design [180,181], website [182], book covers [183], magazine design [92], and

poster design [184,185], among others. Early research on design layouts primarily utilized templates, exemplars, and heuristic design rules [186–189]. These methods, which often required professional design knowledge, leveraged predefined templates or heuristic-based rules but struggled to address the diversity and complexity of design requirements effectively. Subsequent developments introduced techniques such as saliency maps [190] and attention mechanisms [191]. These methods were designed to assess the visual importance within graphic designs, track user attention, and enhance understanding of how users engage with visual elements, marking a significant step towards understanding the dynamics of visual interaction in layouts. Neural networks enable researchers to derive design principles from extensive datasets. CanvasVAE [123] introduced a VAE-based architecture for unconditionally generating vector graphic documents. Following this, LayoutGAN++ [192] further refined this approach by incorporating user-specified constraints into layout generation. LayoutDM [193] employs DDPM to handle geometric parameters in continuous spaces while introducing category information as a condition. LayoutDiffusion [194] treats both geometric parameters and category information as discrete data. LDGM [195] proposes to decouple the diffusion processes to improve the diversity of training samples and learn the reverse process jointly. These methods are learned from the vector domain. A recent work [196] combines the advantages from both bit vector and raster image spaces by proposing a dual diffusion model for design layout generation.

In addition, layout generation in raster images has evolved into two directions: content-agnostic and content-aware layout generation. Content-agnostic layout generation focuses on generating layouts without a predefined content structure. Techniques such as LayoutVAE [197], which utilizes a VAE, and others employing auto-regressive models [198–200] or diffusion models [201,202] have been prominent. DLT [203] further advances this by integrating discrete and continuous data in a diffusion layout transformer. Content-aware layout generation integrates specific visual and textual content into the layouts, aiming to create more contextually relevant designs. Early innovations include Content-GAN [204], which was the first to combine visual and textual elements. Subsequent models like and ICVT [205] employed transformer-based networks and conditional VAEs, respectively, to enhance content integration. PosterLayout [206] uses a CNN-LSTM network focusing on saliency maps. LayoutDETR [207] leverages a detection transformer approach, integrating GAN and VAE technologies and utilizing pre-trained visual and textual encoders for feature extraction.

Furthermore, layouts, which can be encoded in formats such as XML or JSON, are ideally suited for processing by pre-trained LLMs. To this end, a series of works utilize the paradigm of code generation + LLM [208]. LayoutGPT [3] utilizes in-context visual demonstrations in CSS structures to enhance the visual planning capabilities of GPT-3.5/4 for generating layouts from textual conditions. MuLan [209] iteratively plans the layout of an image by deconstructing the text prompt into a sequence of subtasks with an LLM, then revises the image at each step based on feedback from a vision–language model. TextLap [210] enables users to generate layout designs based on natural-language descriptions. LayoutPrompter [211] introduces a training-free approach by leveraging Retrieval-Augmented Generation to enhance the in-context learning capabilities of GPT [212], dynamically sourcing examples from a dataset. However, this retrieval-centric strategy is limited to open-domain generation. These works often overlook the visual domain features or convert them into hard tokens before inputting them into LLMs, which can result in significant information loss.

4.3.2. Glyph Layout Generation

Wang et al. [213] was the first to propose this task. The synthesized layouts of glyphs must consider fine-grained details, such as avoiding the collision of strokes from different glyphs. Furthermore, the placement trajectories of characters should follow a correct reading order (e.g., left to right and top to bottom for English) and possess diverse styles simultaneously, challenges that non-sequence generation models struggle to handle. To address these issues, the authors introduced a dual-discriminator module designed to capture both the character placement trajectory and the rendered shape of the synthesized text logo. However, it faced challenges in designing layouts for long text sequences, adapting to user-defined constraints, and providing diverse layout designs due to the limited quantity of training data. In response, GLDesigner [214] proposed a vision–language model-based framework that generated content-aware text logo layouts by integrating multimodal inputs with user constraints. That study also included the creation of two extensive text logo datasets, which were five times larger than any existing public datasets. In addition to geometric annotations, such as text masks and character recognition, comprehensive layout descriptions in natural language format were provided to enhance reasoning capabilities. Although that model indeed improved the fidelity of generated visual text, it generally fell short in rendering longer textual elements. Lakhanpal et al. [215] introduced a training-free framework to enhance two-stage generation approaches focusing on generating images with long and rare text sequences.

4.4. Colorization

Image colorization is the process of converting grayscale images, including manga [216], line art [217], sketches [218,219], and grayscale photographs [220], into their full-color versions. Various techniques are employed to guide the colorization process, such as scribbles [221–223], reference images [224–232], color palettes [230,233], and textual descriptions [220,234–236]. Scribbles are used to provide intuitive and spontaneous color hints through freehand strokes. The Two-stage Sketch Colorization [237] incorporates a CNN-based system that first applies preliminary color strokes to the canvas, which are later refined to improve color accuracy and detail. Colorization using reference images involves transferring color schemes from an image with similar elements, scenes, or textures. Methods based on stroke application, or edit propagation, allow users to manually introduce color alterations using strokes that are algorithmically extended across the image based on criteria like color similarity and spatial relationships. These methods are invaluable for targeted color adjustments and preserving the authentic appearance of the image. Developments in this field have introduced neural network-driven techniques that automate edit propagation across comparable image structures [238,239]. Palette-based techniques aim to distill the essential color scheme of an image into a select group of representative colors, thereby reducing and abstracting the rich diversity of colors present. Innovations by Chang et al. involved adapting a K-means clustering algorithm to extract a color palette, which laid the groundwork for later advancements. Palette-based models [240] utilize the selected palette as a stylistic guide to influence the overall color theme of the image. Example-based approaches, or style transfer, utilize existing images as templates to guide the recoloring effort, allowing for the transfer of stylistic color elements from one image to another, a process enhanced through the use of CNNs and GANs [7,8]. With the rise of diffusion models, textual descriptions have become a pivotal tool for image generation and thus play a significant role in image colorization. Text-based guidance employs descriptions of desired color themes, object colors, or mood. ControlNet [236] integrates additional trainable modules into pre-existing text-to-image diffusion models [241], tapping into the inherent capabilities of diffusion models for colorization tasks.

4.5. Summary

SVGs use geometric primitives like Bézier curves, polygons, and lines, making them well suited for representing visual concepts in a structured, scalable format. DiffVG [116] allows seamless transitions between raster and vector images. Research in visual text generation can be divided into three main categories: basic text generation, artistic text generation, and text rendering in natural scene images. (1) Basic text generation primarily deals with font transfer, especially for complex scripts such as Chinese or Korean. The focus extends beyond simple generation to include text segmentation into its constituent parts like radicals and strokes, which helps guide the learning process more effectively. (2) Artistic text generation encompasses two main tasks: artistic style transfer and semantic text generation. Both areas use the same basic generation models but tackle different challenges. Artistic style transfer focuses on separating the style and content of text. Semantic text generation, conversely, involves self-deformation of text to maintain readability and aesthetic appeal during automatic re-layout. (3) The rendering of text in natural scene images is particularly challenging due to the need for high clarity and accuracy in diverse visual contexts. To address these challenges, researchers utilize large pre-trained models like Google's T5 series, which is adept at character perception.

The study of layout is a prominent topic. Traditional template-based methods, prevalent in early research, often struggle to encapsulate design rules effectively. Contemporary training-based approaches are categorized based on the type of data used: bitmap data and raster image data. Additionally, a recent innovative work [196] integrates a dual diffusion model encompassing bitmap and raster images. Furthermore, another significant advancement is the encoding of layouts in formats such as XML or JSON, which are highly compatible with processing by pre-trained LLMs. Recent studies have begun to conceptualize layout generation as a language reasoning or planning task. Further developments include the integration of visual information.

Aesthetic research specifically focuses on recolorization. Traditionally, coloring methods have predominantly utilized rule-based systems complemented by color propagation strategies. CNNs were trained on paired grayscale and color images, facilitating the learning of direct mappings from grayscale inputs to their colorized counterparts. Further developments saw the introduction of GANs. In this setup, a generator network learns to create color images that emulate the real colors found in the dataset, while a discriminator network ensures these colorizations are consistent with the original grayscale images. Despite the success of CNNs, their limitations in capturing long-range dependencies prompted researchers to explore other architectures. The introduction of transformers, known for their ability to handle global contexts, brought new opportunities. The most recent advancements involve diffusion models, which integrate pre-trained models to better understand the semantics of color. These models offer guidance during the colorization process by leveraging learned representations of color and its contextual significance, leading to more nuanced and semantically coherent outputs.

5. Present and Future

In Table 2, we summarize the comparison of technologies in AIGD. In this section, we examine the current trend of AIGD, identifies key challenges, and outline future directions and the prevailing research trend of addressing comprehensive problems through holistic solutions.

Table 2. Comparison of technologies in AIGD.

Technology	Key Features	Strengths	Limitations
Object Recognition	MLLMs, DETR-based detection	High accuracy; text integration	Limited coverage; semantic gaps
SVG Recognition	Graph-matching; YOLaT, GNNs	Effective for vectors; abstraction	Ignores high-level info; GNN limits
OCR	TextBoxes, Mask R-CNN, DETR	Diverse scenes; generative support	Complex post-processing; training issues
Font Recognition	CNNs, SVMs, Font-CLIP, GANs	High accuracy; flexible fonts	Font variability; dataset complexity
Layout Analysis	Top-down, bottom-up, hybrid; transformers	Handles irregular layouts; pixel-based	Computationally heavy; table issues
Color Palettes	Regression, VAEAC, transformers	Dynamic, region-specific	Semantic oversight; suboptimal predictions
Other Attributes	CNNs, DMA-Net; personalized models	Global layout encoding; user-focused	Subjective; lacks benchmarks
SVG Generation	SketchRNN, DeepSVG, VectorFusion; VAEs	Scalable; precise; text-conditioned	Dataset dependency; complex synthesis
Vectorization	PolyFit, LIVE, SAMVG; optimization	Preserves detail; high quality	Boundary alignment; perceptual issues
Artistic Typography	DC-Font, Diff-Font; GANs, diffusion	Flexible; supports complex scripts	Data scarcity; readability trade-offs
Visual Text Rendering	TextDiffuser, GlyphControl; multilingual	Accurate segmentation; interactive	Limited dataset diversity; clarity issues
Automatic Layout	LayoutGAN, LayoutGPT; VAEs, LLMs	Diverse needs; language-driven	Complex layouts; visual feature loss
Glyph Layout	GLDesigner; vision–language models	High fidelity; constraint handling	Long text issues; limited data
Colorization	ControlNet, Palette-based; CNNs, GANs	Automated; semantic-aware	Long-range dependencies; inconsistencies

5.1. Analysis Within Perception Tasks

Perception tasks in AIGD focus on analyzing and interpreting design elements, forming the bedrock for intelligent design systems. Here, technologies such as object recognition using MLLMs and DETR-based detection excel in accuracy by leveraging advanced vision–language integration to precisely identify visual objects in raster images. However, when compared to SVG recognition methods (e.g., YOLaT with Graph Neural Networks), object recognition demonstrates superior adaptability in handling complex, multi-level abstractions in vector graphics, though it often incurs higher computational costs due to the semantic processing overhead, making it less efficient for real-time applications in resource-constrained design tools. In terms of usability, MLLMs enhance designer workflows by providing textual outputs that align with natural language queries, but semantic gaps limit their reliability in nuanced graphic contexts, such as interpreting abstract art.

Text element perception technologies, including OCR variants like TextBoxes [56] and Mask R-CNN [61], offer high efficiency in processing diverse scenes, with generative models supporting quick adaptations to irregular text. Yet, compared to font recognition approaches (e.g., CNNs with FontCLIP [242]), OCR suffers from lower accuracy in noisy or variable environments due to complex post-processing needs, reducing its adaptability for multilingual or stylized fonts in global design projects. Usability is a strength for font recognition, as tools like GANs enable flexible font selection with minimal user intervention, though dataset complexity poses barriers for non-expert designers.

Layout analysis methods (e.g., hybrid top-down/bottom-up with transformers) strike a balance in efficiency by managing irregular layouts through pixel-based extraction, outperforming specialized perception tools in adaptability to complex documents like infographics. However, their computational intensity hampers usability in everyday graphic design software, where faster alternatives like Faster-RCNN might be preferred despite occasional inaccuracies in table handling.

Aesthetic understanding technologies reveal stark contrasts: color palette recommendation (e.g., VAEAC [243] and transformers) provides dynamic, region-specific outputs with high efficiency for iterative design but overlooks semantics, leading to suboptimal accuracy compared to other attribute models (e.g., DMA-Net), which encode global layouts and user perceptions more adaptively. Usability favors personalized models, yet the subjective nature of aesthetics lacks standardized benchmarks, making cross-technology comparisons challenging in collaborative design environments.

Overall, perception technologies prioritize accuracy and adaptability in structured tasks but lag in efficiency for high-volume design workflows, with usability often compromised by training dependencies.

5.2. Analysis Within Generation Tasks

Generation tasks shift toward creating new design elements, where AI's creative potential shines but introduces variability in performance metrics. Non-text element generation, exemplified by SVG generation (e.g., DeepSVG with VAEs and diffusion models), achieves high scalability and precise control, offering superior efficiency over vectorization techniques (e.g., PolyFit with optimization), which preserve details but struggle with boundary alignment, reducing accuracy in perceptual fidelity. Adaptability is a key differentiator: text-conditioned SVG models like VectorFusion excel in diverse graphic styles, enhancing usability for designers experimenting with AI-assisted ideation, though dataset dependencies limit their robustness in novel scenarios.

In text element generation, artistic typography (e.g., Diff-Font with GANs) provides flexibility for complex scripts, surpassing visual text rendering (e.g., TextDiffuser [244], TextDiffuser-2 [166], Artist [245]) in adaptability to multilingual designs, but at the cost of readability trade-offs that diminish accuracy. Efficiency favors diffusion-based rendering due to interactive segmentation, improving usability in real-time editing tools, yet data scarcity hampers both in underrepresented languages.

Layout generation technologies, such as automatic layouts with LayoutGAN [246] and LLMs, handle diverse requirements efficiently via natural language inputs, offering greater usability than glyph layouts (e.g., GLDesigner [214]), which provide high fidelity but falter with long texts. Accuracy in constraint handling makes glyph models more adaptable for precise typography, though visual feature loss in automatic methods reduces overall effectiveness in multilayered compositions.

Aesthetic refinement, particularly colorization (e.g., ControlNet [236] with GANs), automates adjustments with semantic awareness, excelling in efficiency and usability for

guided edits, but long-range dependencies lead to inconsistencies, lowering accuracy compared to palette-based approaches in scene-specific tasks.

Generation technologies generally outperform perception ones in adaptability and usability by enabling creative outputs, yet they demand higher efficiency in handling dependencies, with accuracy often traded for speed in generative processes.

5.3. MLLM for Graphic Design

Across perception and generation, a clear synergy emerges: perception technologies like MLLMs feed into generation tasks (e.g., text-conditioned SVG), enhancing overall accuracy in end-to-end workflows. However, efficiency gaps persist—perception's computational demands slow generation in integrated systems, while adaptability favors multimodal approaches that bridge visual and semantic gaps. Usability is amplified in 2025 trends like AI realism and personalization, where tools such as Runway ML automate repetitive tasks, allowing designers to focus on empathy and originality. Challenges remain in interpretability and human intent alignment, with generative AI risking over-automation that erodes designer control. In graphic design contexts, these comparisons reveal AI as an augmentative force rather than a replacement, boosting efficiency in mundane tasks while demanding human oversight for adaptability in creative, user-centric projects. Future directions should prioritize hybrid models integrating LLMs for better intent understanding, alongside ethical considerations for bias in datasets, to foster more inclusive and usable AIGD ecosystems. We provide a pseudocode example framework that handles multimodal inputs and illustrates the methodology behind integrating text and visual prompts. Pseudocode can be found in Listing 1 below.

Listing 1. Pseudocode for Multimodal Input Processing with Practical Integration.

```
# Step 1: Process Text Input
from transformers import AutoTokenizer, AutoModel

# Load a pretrained text model (e.g., BERT or CLIP's text encoder)
text_tokenizer = AutoTokenizer.from_pretrained("openai/clip-vit-base-patch32")
text_model = AutoModel.from_pretrained("openai/clip-vit-base-patch32")

text_input = "Create␣a␣modern␣poster␣with␣a␣blue␣theme."
text_tokens = text_tokenizer(text_input, return_tensors="pt", truncation=True, padding=True)
text_embedding = text_model(**text_tokens).last_hidden_state.mean(dim=1)  # Extract mean~pooling

# Step 2: Process Visual Input
import cv2
import torch
from torchvision import~transforms

# Load the visual input (reference image)
visual_input = cv2.imread("reference_image.jpg")
# Preprocess the image (resize, normalize)
transform = transforms.Compose([
    transforms.ToPILImage(),
    transforms.Resize((224, 224)),
    transforms.ToTensor(),
```

```
    transforms.Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225])
])
visual_input_tensor = transform(visual_input).unsqueeze(0)

# Load a pretrained visual model (e.g., CLIP's vision encoder)
visual_model = AutoModel.from_pretrained("openai/clip-vit-base-patch32")
visual_features = visual_model.get_image_features(visual_input_tensor)

# Step 3: Multimodal Fusion
# Combine text and visual embeddings (e.g., concatenation followed by linear projection)
combined_representation = torch.cat((text_embedding, visual_features), dim=1)

# Optionally pass through a feedforward network for improved fusion
from torch import nn
fusion_layer = nn.Sequential(
    nn.Linear(combined_representation.shape[1], 512),
    nn.ReLU(),
    nn.Linear(512, 256)
)
fused_representation = fusion_layer(combined_representation)

# Step 4: Generate Design
# Use the fused representation to generate a design (e.g., via a GAN or diffusion model)
from some_design_library import DesignGenerator  # Replace with actual~implementation

design_generator = DesignGenerator()
design = design_generator.generate(fused_representation)

# Save the output design
cv2.imwrite("final_design.jpg", design)
```

Recent advancements in multimodal LLMs have shown promising applications in graphic design tasks. As shown in Figure 4, these approaches can be categorized into several key directions based on their architectural design and application focus.

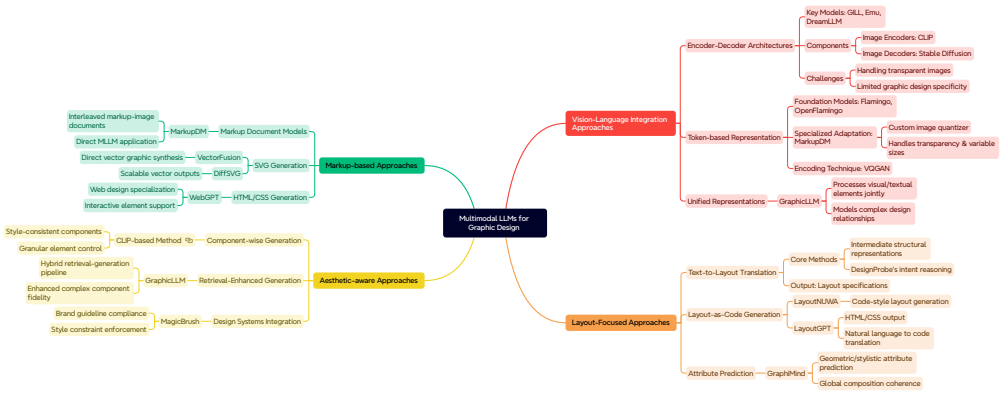

Figure 4. Overview of existing works in multimodal LLM for graphic design.

Vision–language integration approaches connect LLMs with pre-trained image components:

- Encoder–Decoder Architectures: Models such as DreamLLM [247] integrate LLMs with pre-trained image encoders (e.g., CLIP [171]) and decoders (e.g., Stable Diffusion [241]). While powerful for general image generation, these approaches face challenges with transparent images common in graphic design [248]. OpenCLOE [249] begins by translating user intentions into a design plan using GPT-3.5 and in-context learning. Then, the image and typography generation modules synthesize design elements according to the specified plan, and the graphic renderer assembles the final image.
- Token-based Representation: An alternative approach represents images as discrete tokens [10,250,251]. This method encodes images into token sequences via image quantizers like VQGAN [252]. The MarkupDM approach [248] adapts this methodology specifically for graphic design by developing a custom quantizer that handles transparency and varying image sizes.
- Unified Models: GraphicLLM [253] proposes a multimodal model that processes both visual and textual design elements within a unified framework, addressing the complex relationships.

Layout-focused approaches leverage LLMs specifically for layout generation tasks:

- Text-to-Layout Translation: The authors of [254] utilize LLMs to translate descriptions into intermediate structural representations that guide subsequent layout generation. DesignProbe [19] extends this by introducing a reasoning mechanism where LLMs analyze design intent before generating structured layout specifications.
- Layout-as-Code Generation: LayoutNUWA [208] treats layout generation as a code generation task, leveraging the programming capabilities of LLMs. Similarly, LayoutGPT [3] functions as a layout generator by producing HTML/CSS code from textual prompts.
- Attribute Prediction: GraphiMind [25] employs MLLMs to predict geometric and stylistic attributes for design elements while maintaining global coherence across the entire composition.

Aesthetic-aware approaches aim to address multiple aspects of design simultaneously:

- Component-wise Generation: The authors of [255] propose a method that leverages CLIP embeddings to generate design components that maintain stylistic consistency. VASCAR [256] is large vision–language model-based content-aware layout generation. Design-o-meter [257] is the first work to score and refine designs within a unified framework by adjusting the layout of design elements to achieve high aesthetic scores.
- Retrieval-Enhanced Generation: GraghicLLM [253] combines generative capabilities with retrieval mechanisms to leverage existing design elements, achieving higher fidelity results for complex graphic components.
- Design Systems Integration: MagicBrush [258] integrates with design systems to ensure generated elements conform to established brand guidelines and stylistic constraints.

Markup-based approaches involve representing designs as markup language:

- Markup Document Models: MarkupDM [248] introduces a novel approach treating graphic designs as interleaved multimodal documents consisting of markup language and images. This representation allows direct application of multimodal LLMs to graphic design tasks.
- SVG Generation: VectorFusion [126] focus on generating vector graphics (SVG) directly, addressing the scalability advantages needed for professional graphic design workflows.

- HTML/CSS Generation: WebGPT [259] generates web-based designs by producing HTML and CSS code, demonstrating the potential of code-centric approaches for interactive designs.

MLLMs such as LayoutGPT represent a significant advancement in automated layout and graphic design. However, their performance is often evaluated using a combination of general-purpose and task-specific metrics, including the following: BLEU measures the similarity of generated layouts to reference layouts, particularly in structured tasks like table or form generation; Fréchet Inception Distance (FID) evaluates the visual quality of generated outputs by comparing them to real-world designs, measuring how realistic and well aligned the results are; and Alignment Accuracy quantifies the placement of elements (e.g., text, images, buttons) with respect to design guidelines or user-defined constraints.

Traditional methods often rely on predefined templates and rules, which limit flexibility but ensure quick execution for specific tasks. In contrast, LayoutGPT leverages generative capabilities to create custom layouts dynamically, enabling scalability across various design domains. While traditional methods ensure consistent adherence to design principles, they may struggle with complex or non-standard layouts. LayoutGPT, powered by deep learning, generates visually diverse and highly creative designs but may occasionally produce outputs that require manual refinement. Multimodal LLMs like LayoutGPT excel in automating the design process, reducing manual effort through intelligent suggestions and rapid prototyping. Unlike rule-based systems, they adapt to user-specific requirements and diverse input modalities, including text, images, and sketches. A case study can be found in Listing 2 below.

Listing 2. Case study: traditional template-based graphic design.

```
# Step 1: Select a pre-existing template
template = select_template(tool="Canva")

# Step 2: Fill in content
headline = "Try␣Our␣New␣Pumpkin␣Spice␣Latte!"
image = "stock_image_of_coffee.jpg"
footer = {
    "logo": "coffee_shop_logo.png",
    "contact": "123␣Coffee␣Street,␣Brewtown",
    "social": "@coffeelovers"
}

# Step 3: Apply predefined styles
apply_styles(
    layout="fixed",
    color_scheme="predefined",
    typography="default"
)

# Step 4: Template Features
layout = {
    "headline": "Top␣of␣the␣flyer",
    "image": "Center␣of␣the␣flyer",
    "footer": "Bottom␣of␣the␣flyer"
}
constraints = {
    "color_scheme": "Limited",
    "typography": "Static"
}
```

```
# Step 5: Generate flyer output
flyer = {
    "headline": "Try Our New Pumpkin Spice Latte!",
    "image": "stock_image_of_coffee.jpg",
    "footer": {
        "logo": "coffee_shop_logo.png",
        "contact": "123 Coffee Street, Brewtown",
        "social": "@coffeelovers"
    }
}
```

5.4. Existing Challenges

Despite remarkable advances in AIGD, significant technical barriers that limit practical applications persist. As illustrated in Figure 5, conventional approaches suffer from three fundamental limitations: inadequate user intention understanding, limited interpretability, and insufficient layer control. These challenges reflect deeper systemic issues within AI architectures requiring targeted research attention.

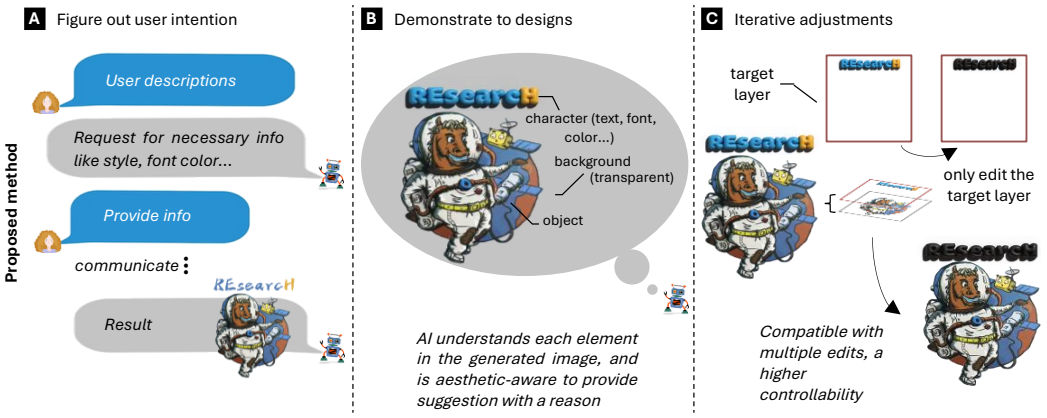

Figure 5. Demonstration of AIGD characteristics, including autonomy in (**A**) to figure out user intention, interpretability in (**B**) where the explanation can be provided to demonstrate designs, and multilayered aspect in (**C**) when editing generated designs.

The problem of inadequate user intention understanding (Figure 5A) represents a fundamental cognitive–computational gap in current systems. While text-to-image models have advanced significantly, they fundamentally operate through statistical pattern matching rather than the semantic understanding of design requirements. This semantic–representational decoupling manifests when converting textual briefs to visual styles, where the design intention encoding $\mathbf{h}I = \text{LLM}\theta(\mathbf{x}\text{text}) \oplus \text{ViT}\phi(\mathbf{x}_{\text{image}})$ yields demonstrably lower style consistency than human designers in controlled trials. The challenge extends beyond simple prompt engineering to the more complex problem of modeling designers' cognitive processes when translating abstract requirements into concrete visual decisions. Current multimodal approaches attempt to bridge this gap through joint embeddings but fail to capture the nuanced contextual reasoning that experienced designers apply to interpret client needs, audience expectations, and brand guidelines simultaneously [248,253]. This intention–representation gap becomes particularly evident in iterative feedback scenarios, where AI systems struggle to incorporate targeted revisions without regenerating entire compositions. This limitation significantly reduces their utility.

The interpretability challenge (Figure 5B) reflects a deeper epistemological problem in AI-driven design: the inability to articulate design rationale in terms that align with established design principles and practices. Current systems provide generic explanations that lack specificity regarding composition decisions, color harmony, typographic choices, and other critical design elements. This limitation stems partly from how design knowledge is encoded during training—predominantly through implicit pattern recognition rather than explicit design theory. The catastrophic performance degradation when transitioning from atomic to composite tasks represents a direct consequence of this interpretability deficit. Models cannot effectively decompose complex tasks into meaningful subproblems or explain the interrelationships between design elements without an explicit representational framework for design principles. Recent work in design rationale extraction has attempted to explain generated designs retroactively. Still, these post hoc rationalizations often fail to reflect the actual generative process or provide actionable insights for refinement. Typography presents an additional interpretability challenge—while font generation models have achieved impressive stylistic accuracy, they frequently fail to balance aesthetic considerations with functional requirements like readability across contexts, proper kerning, and linguistic nuance.

The multiple layers and iterative editing problem (Figure 5C) reveals fundamental architectural limitations in current generative models when applied to professional graphic design workflows. Unlike the photography-oriented generation, graphic design requires precise control over individual elements, relationships, and layer hierarchies. Conventional methods struggle with layer-specific editing—modifications to one element often unintentionally affect others, as evident in tools like TurboEdit and Flux.1. This limitation stems from pre-trained image encoders and decoders that inadequately support transparent images and multilayered compositions [248]. The technical challenge extends beyond simple transparency support to the more complex problem of maintaining semantic consistency across layers while enabling targeted modifications. Standard diffusion models and transformers operate on flattened representations that fail to preserve the logical independence of design elements. This represents a fundamental tension between maintaining high-fidelity visual details and producing clean, scalable vector representations—a trade-off that simultaneously impacts editability, rendering performance, and file size.

Beyond these three primary challenges, a fourth systemic limitation has emerged in recent research: contextual consistency maintenance across design artifacts. Professional graphic design rarely involves isolated images, instead requiring coherent visual systems spanning multiple formats and applications while maintaining brand identity. Current AI approaches treat each generation as an independent task, lacking the architectural components to model and maintain cross-artifact consistency. This limitation becomes particularly problematic in comprehensive design systems where visual elements must adapt to different contexts (responsive web design, print media, environmental applications) while preserving core identity elements. The computational challenge involves maintaining a persistent design representation that can flexibly adapt to new constraints without sacrificing fundamental stylistic principles—a problem that remains largely unaddressed in current research. These challenges collectively point to a need for more sophisticated architectural approaches that better align with the cognitive processes, theoretical foundations, and practical workflows of professional graphic design. While recent multimodal models show promising capabilities in specific domains, bridging the gap to comprehensive design assistance requires addressing these fundamental limitations through targeted research in representation learning, explainable generative processes, compositional reasoning, and layered editing paradigms.

5.5. Potential Directions

Recent trends in AI for graphic design suggest several promising research directions that warrant further investigation. This section explores potential avenues for advancing the field, focusing on developing unified approaches and addressing specific challenges in design understanding and generation.

Towards Unified End-to-End Models. Recent advances in MLLMs demonstrate the feasibility of employing unified end-to-end solutions [260] for AIGD tasks. These models would integrate multimodal intent understanding, high-quality visual element generation, and knowledge-enhanced layout reasoning within a single framework. Such integration aligns with current academic trajectories in multimodal learning and offers a promising pathway for comprehensively addressing the complex challenges of graphic design automation.

- Multimodal Intent Understanding. Current multimodal models integrating dialogue and visual recognition provide a foundation for intent understanding but require significant enhancement in several key areas: (1) Graphic design presents unique challenges with artistic images featuring diverse fonts and complex layouts that exceed the capabilities of general-purpose recognition systems. (2) Three-dimensional designs, text with special effects (overlapping, bending, distortion), and artistic typography demand specialized recognition approaches. (3) Enhanced communicative abilities in large language models are needed to translate ambiguous user inputs into coherent, actionable design specifications.
- Knowledge-Enhanced Layout Reasoning. The computational representation of abstract design principles presents significant challenges. Drawing inspiration from advanced reasoning models like OpenAI o1, research should focus on the following: (1) Encoding established design theories within computational frameworks. (2) Developing inference mechanisms that can apply these principles contextually. (3) Creating evaluation metrics that align with human aesthetic judgment. (4) Building models that can explain their layout decisions with reference to design principles.
- High-Quality Visual Element Generation. Layer diffusion techniques show promise for creating images with transparent backgrounds—a critical requirement for graphic design. However, text generation capabilities require substantial improvement, particularly for artistic typography, where models like Flux.1 demonstrate potential but insufficient fidelity. Meanwhile, LLM-guided approaches for generating vector graphics, exemplified by tools like SVGDreamer, offer precision and scalability advantages. Research should focus on enhancing text rendering and incorporating deeper reasoning about design principles. Finally, models capable of a seamless transition between raster and vector formats could revolutionize workflow efficiency by offering the advantages of both paradigms, as suggested by [196].

Research in Sub-directions. Beyond unified models, several specialized research directions show particular promise: (1) Developing encoders specifically trained on graphic design elements could substantially improve the representation of design-specific features. Unlike general-purpose visual encoders, design-specific approaches would prioritize typographical feature representation, layout structure encoding, color harmony, and palette relationships. (2) Interactive and collaborative design systems enable iterative refinement and feedback loops between the designer and AI, focusing on turn-taking mechanisms for collaborative design, interpretable design suggestions, learning from designer feedback, and preserving creative agency while enhancing productivity. (3) Design rationale understanding models that capture the underlying reasoning in design decisions, rather than just visual patterns, represent a critical frontier [19], which involves inferring design intentions from examples, reverse-engineering design decisions, representing the relation-

ship between design goals and visual implementations, and learning from design critique and evaluation. (4) Improved mechanisms for transferring design styles between different modalities offer significant potential for design consistency and efficiency [22]. (5) The vector graphics domain, particularly SVG, remains underexplored despite its importance in graphic design. Recent work by [261] introduces Primal Visual Description (PVD), a textual representation that translates SVG into abstractions comprising primitive attributes (shape, position, measurement) and their values. Research in this domain should explore the integration of these structured representations with generative and reasoning capabilities, potentially offering precision advantages over raster-based approaches for graphic designs.

The intersection of AI and graphic design presents rich research opportunities that bridge visual generation, multimodal understanding, and design reasoning. The emerging field of multimodal LLMs for graphic design demonstrates significant promise for automating and enhancing design workflows, particularly as these models continue to improve in handling the unique characteristics of design documents, including transparency, layout constraints, and stylistic coherence. Progress will require interdisciplinary approaches to transform design workflows while preserving and enhancing human creative agency.

Potentials in Other Practical Applications. While this survey primarily focused on AIGD methods for vector graphics and typography, the methodologies discussed are highly adaptable and extendable to other domains, such as UI/UX design and print media. For example, in the domain of UI/UX design, AIGD methods such as automated layout generation have been successfully implemented to optimize interface design. Case studies illustrate how AI-driven layout tools suggest user-friendly arrangements of buttons, menus, and other UI components. These methods leverage constraints like usability heuristics and accessibility guidelines to create dynamic layouts that adapt to diverse screen sizes and user preferences. Similarly, Adobe XD's AI-powered tools demonstrate how AI can assist designers in decision-making processes, such as selecting color schemes, typography, and interaction patterns based on user behavior data and cognitive load models. Generative design techniques, such as those employed by Figma's Variants feature for prototyping, enable rapid iterations by producing multiple design variations for A/B testing or user feedback collection. The principles explored in this survey, particularly for typography and vector graphics, also find direct applications in print media. For instance, AI's role in creating consistent typographic hierarchies and vector-based layouts that scale seamlessly across different print formats has been well documented. A notable case study is Canva's AI-powered design assistant, which helps users create brochures, posters, and packaging designs that align with brand guidelines. Similarly, Coca-Cola's use of AI-generated packaging designs highlights how AIGD can produce visually cohesive and on-brand materials for mass production. Furthermore, AI-driven tools like Adobe InDesign's Liquid Layouts have enabled the creation of personalized print materials, such as tailored advertisements or invitations, by combining user data with generative design techniques. The underlying algorithms for layout generation, typography optimization, and style synthesis are domain-agnostic, as demonstrated in studies like Microsoft's Project Trove, which facilitates seamless transitions between digital (UI/UX) and physical (print) design contexts. By incorporating domain-specific constraints—such as print resolution for physical media or screen responsiveness for digital outputs—AIGD methods have been shown to support diverse design workflows across industries.

6. Conclusions

This survey comprehensively reviewed AI's state-of-the-art methods and applications in graphic design, categorizing them into perception and generation tasks. We explored various subtasks, including non-text element perception, text element perception, layout

analysis, aesthetic understanding, and the generation of non-text elements, text elements, layouts, and colors. Integrating large language models and multimodal approaches has become a pivotal trend, enabling more holistic and context-aware design solutions. However, several challenges persist, such as the need to better understand human intent, improved interpretability of AI-generated designs, and enhanced control over multilayered compositions. Future research should develop unified end-to-end models integrating multimodal understanding, high-fidelity generation, and design reasoning.

Author Contributions: Conceptualization, X.Z. and W.Z.; methodology, X.Z.; validation, X.Z., W.Z. and N.Z.; formal analysis, X.Z.; investigation, W.Z.; data curation, N.Z.; writing—original draft preparation, X.Z.; writing—review and editing, X.Z., W.Z. and N.Z.; visualization, W.Z.; project administration, W.Z.; funding acquisition, X.Z. All authors have read and agreed to the published version of the manuscript.

Funding: The work described in this paper was fully/substantially/partially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. PolyU/RGC Project PolyU 25211424).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created or analyzed in this study.

Conflicts of Interest: Author Nanxuan Zhao was employed by the company Adobe Inc. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Jiang, S.; Wang, Z.; Hertzmann, A.; Jin, H.; Fu, Y. Visual font pairing. *IEEE Trans. Multimed.* **2019**, *22*, 2086–2097. [CrossRef]
2. Weng, H.; Huang, D.; Zhang, T.; Lin, C.Y. Learn and Sample Together: Collaborative Generation for Graphic Design Layout. In Proceedings of the International Joint Conference on Artificial Intelligence, Macao, China, 19–25 August 2023; pp. 5851–5859.
3. Feng, W.; Zhu, W.; Fu, T.J.; Jampani, V.; Akula, A.; He, X.; Basu, S.; Wang, X.E.; Wang, W.Y. Layoutgpt: Compositional visual planning and generation with LLMs. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 10–15 December 2024; Volume 36, pp. 18225–18250.
4. Tan, J.; Lien, J.M.; Gingold, Y. Decomposing images into layers via RGB-space geometry. *ACM Trans. Graph. (TOG)* **2016**, *36*, 1–14. [CrossRef]
5. Tan, J.; Echevarria, J.; Gingold, Y. Efficient palette-based decomposition and recoloring of images via RGBXY-space geometry. *ACM Trans. Graph. (TOG)* **2018**, *37*, 1–10. [CrossRef]
6. Wang, Y.; Liu, Y.; Xu, K. An improved geometric approach for palette-based image decomposition and recoloring. In Proceedings of the International Joint Conference on Artificial Intelligence, Porto, Portugal, 3–7 June 2019; pp. 11–22.
7. Zhang, R.; Isola, P.; Efros, A.A. Colorful image colorization. In *Computer Vision—ECCV 2016. ECCV 2016*; Springer: Cham, Switzerland, 2016; pp. 649–666.
8. Vitoria, P.; Raad, L.; Ballester, C. Adversarial picture colorization with semantic class distribution. In Proceedings of the Winter Conference on Applications of Computer Vision, Snowmass Village, CO, USA, 11–14 October 2020; pp. 45–54.
9. Huang, D.; Guo, J.; Sun, S.; Tian, H.; Lin, J.; Hu, Z.; Lin, C.Y.; Lou, J.G.; Zhang, D. A survey for graphic design intelligence. *arXiv* **2023**, arXiv:2309.01371. [CrossRef]
10. Liu, H.; Li, C.; Wu, Q.; Lee, Y.J. Visual instruction tuning. In Proceedings of the Advances in Neural Information Processing Systems, New Orleans, LA, USA, 10–16 December 2023; Volume 36, pp. 34892–34916.
11. Hu, H.; Chan, K.C.; Su, Y.C.; Chen, W.; Li, Y.; Sohn, K.; Zhao, Y.; Ben, X.; Gong, B.; Cohen, W.; et al. Instruct-Imagen: Image generation with multi-modal instruction. In Proceedings of the Advances in Neural Information Processing Systems, Seattle, WA, USA, 17–21 June 2024; Volume 36, pp. 4754–4763.
12. Epstein, Z.; Hertzmann, A.; Investigators of Human Creativity; Akten, M.; Farid, H.; Fjeld, J.; Frank, M.R.; Groh, M.; Herman, L.; Leach, N.; et al. Art and the science of generative AI. *Science* **2023**, *380*, 1110–1111. [CrossRef] [PubMed]
13. Tian, X.; Günther, T. A survey of smooth vector graphics: Recent advances in representation, creation, rasterization and image vectorization. *IEEE Trans. Vis. Comput. Graph.* **2022**, *30*, 1652–1671. [CrossRef]

14. Shi, Y.; Shang, M.; Qi, Z. Intelligent layout generation based on deep generative models: A comprehensive survey. *Inf. Fusion* **2023**, *100*, 140. [CrossRef]
15. Liu, Z.; Liu, F.; Zhang, M. Intelligent Graphic Layout Generation: Current Status and Future Perspectives. In Proceedings of the International Conference on Computer Supported Cooperative Work in Design, Tianjin, China, 8–10 May 2024; pp. 2632–2637.
16. Tang, Y.; Ciancia, M.; Wang, Z.; Gao, Z. What's Next? Exploring Utilization, Challenges, and Future Directions of AI-Generated Image Tools in Graphic Design. *arXiv* **2024**, arXiv:2406.13436.
17. Tang, Y.; Ciancia, M.; Wang, Z.; Gao, Z. Vision-language models for vision tasks: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 5625–5644. [CrossRef]
18. Li, J.; Li, D.; Xiong, C.; Hoi, S. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 17–23 July 2022; pp. 12888–12900.
19. Lin, J.; Huang, D.; Zhao, T.; Zhan, D.; Lin, C.Y. DesignProbe: A Graphic Design Benchmark for Multimodal Large Language Models. *arXiv* **2024**, arXiv:2404.14801. [CrossRef]
20. Cheng, Y.; Zhang, Z.; Yang, M.; Nie, H.; Li, C.; Wu, X.; Shao, J. Graphic Design with Large Multimodal Model. *arXiv* **2024**, arXiv:2404.14368. [CrossRef]
21. Xiao, S.; Wang, Y.; Zhou, J.; Yuan, H.; Xing, X.; Yan, R.; Li, C.; Wang, S.; Huang, T.; Liu, Z. Omnigen: Unified image generation. *arXiv* **2024**, arXiv:2409.11340. [CrossRef]
22. Zhou, C.; Yu, L.; Babu, A.; Tirumala, K.; Yasunaga, M.; Shamis, L.; Kahn, J.; Ma, X.; Zettlemoyer, L.; Levy, O. Transfusion: Predict the next token and diffuse images with one multi-modal model. *arXiv* **2024**, arXiv:2408.11039. [CrossRef]
23. Meggs, P.B. *Type and Image: The Language of Graphic Design*; John Wiley & Sons: Hoboken, NJ, USA, 1992.
24. Dou, S.; Jiang, X.; Liu, L.; Ying, L.; Shan, C.; Shen, Y.; Dong, X.; Wang, Y.; Li, D.; Zhao, C. Hierarchical Recognizing Vector Graphics and A New Chart-based Dataset. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 7556–7573. [CrossRef]
25. Huang, Q.; Lu, M.; Lanir, J.; Lischinski, D.; Cohen-Or, D.; Huang, H. GraphiMind: LLM-centric Interface for Information Graphics Design. *arXiv* **2024**, arXiv:2401.13245.
26. Ding, S.; Chen, X.; Fang, Y.; Liu, W.; Qiu, Y.; Chai, C. DesignGPT: Multi-Agent Collaboration in Design. In Proceedings of the International Symposium on Computational Intelligence and Design, Hangzhou, China, 16–17 December 2023; pp. 204–208.
27. Weng, H.; Huang, D.; Qiao, Y.; Hu, Z.; Lin, C.Y.; Zhang, T.; Chen, C.L. Desigen: A Pipeline for Controllable Design Template Generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 17–21 June 2024; pp. 21–32.
28. Luo, W.; Zhang, H.; Li, J.; Wei, X.S. Learning semantically enhanced feature for fine-grained image classification. *IEEE Signal Process. Lett.* **2020**, *27*, 1545–1549. [CrossRef]
29. O'Gorman, L. The document spectrum for page layout analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **1993**, *15*, 1162–1173. [CrossRef]
30. Long, S.; Qin, S.; Panteleev, D.; Bissacco, A.; Fujii, Y.; Raptis, M. Towards end-to-end unified scene text detection and layout analysis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 1049–1059.
31. Cheng, H.; Zhang, P.; Wu, S.; Zhang, J.; Zhu, Q.; Xie, Z.; Li, J.; Ding, K.; Jin, L. M6doc: A large-scale multi-format, multi-type, multi-layout, multi-language, multi-annotation category dataset for modern document layout analysis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 15138–15147.
32. Luo, C.; Shen, Y.; Zhu, Z.; Zheng, Q.; Yu, Z.; Yao, C. LayoutLLM: Layout Instruction Tuning with LLMs for Document Understanding. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 17–21 June 2024; pp. 15630–15640.
33. Chen, Y.; Zhang, J.; Peng, K.; Zheng, J.; Liu, R.; Torr, P.; Stiefelhagen, R. RoDLA: Benchmarking the Robustness of Document Layout Analysis Models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 17–21 June 2024; pp. 15556–15566.
34. Zhang, N.; Cheng, H.; Chen, J.; Jiang, Z.; Huang, J.; Xue, Y.; Jin, L. M2Doc: A Multi-Modal Fusion Approach for Document Layout Analysis. In Proceedings of the Association for the Advancement of Artificial Intelligence, London, UK, 17–19 October 2024; pp. 7233–7241.
35. Chen, R.; Cheng, J.K.; Ma, J. A Fusion Framework of Whitespace Smear Cutting and Swin Transformer for Document Layout Analysis. In Proceedings of the International Conference on Intelligent Computing, Tianjin, China, 5–8 August 2024; pp. 338–353.
36. Kong, Y.; Luo, C.; Ma, W.; Zhu, Q.; Zhu, S.; Yuan, N.; Jin, L. Look closer to supervise better: One-shot font generation via component-based discriminator. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 13482–13491.
37. He, H.; Chen, X.; Wang, C.; Liu, J.; Du, B.; Tao, D.; Yu, Q. Diff-font: Diffusion model for robust one-shot font generation. *Int. J. Comput. Vis.* **2024**, *132*, 5372–5386. [CrossRef]

38. Tang, L.; Cai, Y.; Liu, J.; Hong, Z.; Gong, M.; Fan, M.; Han, J.; Liu, J.; Ding, E.; Wang, J. Few-shot font generation by learning fine-grained local styles. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 7895–7904.
39. Chen, L.; Lee, F.; Chen, H.; Yao, W.; Cai, J.; Chen, Q. Automatic Chinese font generation system reflecting emotions based on generative adversarial network. *Appl. Sci.* **2020**, *10*, 5976. [CrossRef]
40. Wang, C.; Zhou, M.; Ge, T.; Jiang, Y.; Bao, H.; Xu, W. Cf-font: Content fusion for few-shot font generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 1858–1867.
41. Wang, Y.; Gao, Y.; Lian, Z. Attribute2font: Creating fonts you want from attributes. *ACM Trans. Graph.* **2020**, *39*, 69. [CrossRef]
42. Liu, W.; Liu, F.; Ding, F.; He, Q.; Yi, Z. Xmp-font: Self-supervised cross-modality pre-training for few-shot font generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 7905–7914.
43. Yan, S. ReDualSVG: Refined Scalable Vector Graphics Generation. In *International Conference on Artificial Neural Networks*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 87–98.
44. Cao, D.; Wang, Z.; Echevarria, J.; Liu, Y. Svgformer: Representation learning for continuous vector graphics. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 10093–10102.
45. Zhao, Z.; Chen, Y.; Hu, Z.; Chen, X.; Ni, B. Vector Graphics Generation via Mutually Impulsed Dual-domain Diffusion. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 17–21 June 2024; pp. 4420–4428.
46. Podell, D.; English, Z.; Lacey, K.; Blattmann, A.; Dockhorn, T.; Müller, J.; Penna, J.; Rombach, R. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv* **2023**, arXiv:2307.01952. [CrossRef]
47. Singh, J.; Gould, S.; Zheng, L. High-fidelity guided image synthesis with latent diffusion models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 5997–6006.
48. Cheng, G.; Yuan, X.; Yao, X.; Yan, K.; Zeng, Q.; Xie, X.; Han, J. Towards large-scale small object detection: Survey and benchmarks. *IEEE Tran. Pattern Anal. Mach. Intell.* **2023**, *45*, 13467–13488. [CrossRef]
49. Chen, J.; Guo, H.; Yi, K.; Li, B.; Elhoseiny, M. Visualgpt: Data-efficient adaptation of pretrained models for image captioning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 18030–18040.
50. Wan, J.; Song, S.; Yu, W.; Liu, Y.; Cheng, W.; Huang, F.; Bai, X.; Yao, C.; Yang, Z. OmniParser: A Unified Framework for Text Spotting Key Information Extraction and Table Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 17–21 June 2024; pp. 15641–15653.
51. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 213–229.
52. Locteau, H.; Adam, S.; Trupin, E.; Labiche, J.; Héroux, P. Symbol spotting using full visibility graph representation. In Proceedings of the Graphics Recognition, Curitiba, Brazil, 20–21 September 2007; pp. 49–50.
53. Ramel, J.Y.; Vincent, N.; Emptoz, H. A structural representation for understanding line-drawing images. *Doc. Anal. Recognit.* **2000**, *3*, 58–66. [CrossRef]
54. Jiang, X.; Liu, L.; Shan, C.; Shen, Y.; Dong, X.; Li, D. Recognizing vector graphics without rasterization. In Proceedings of the Advances in Neural Information Processing Systems, Virtual, 6–14 December 2021; Volume 34, pp. 24569–24580.
55. Bi, T.; Zhang, X.; Zhang, Z.; Xie, W.; Lan, C.; Lu, Y.; Zheng, N. Text Grouping Adapter: Adapting Pre-trained Text Detector for Layout Analysis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 17–21 June 2024; pp. 28150–28159
56. Liao, M.; Shi, B.; Bai, X.; Wang, X.; Liu, W. Textboxes: A fast text detector with a single deep neural network. In Proceedings of the Association for the Advancement of Artificial Intelligence Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 31.
57. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
58. Tian, Z.; Huang, W.; He, T.; He, P.; Qiao, Y. Detecting text in natural image with connectionist text proposal network. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 56–72.
59. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Tran. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
60. Liu, Y.; Shen, C.; Jin, L.; He, T.; Chen, P.; Liu, C.; Chen, H. Abcnet v2: Adaptive bezier-curve network for real-time end-to-end text spotting. *IEEE Tran. Pattern Anal. Mach. Intell.* **2021**, *44*, 8048–8064. [CrossRef]
61. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.

62. Meng, D.; Chen, X.; Fan, Z.; Zeng, G.; Li, H.; Yuan, Y.; Sun, L.; Wang, J. Conditional detr for fast training convergence. In Proceedings of the IEEE International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 3651–3660.
63. Wang, Y.; Zhang, X.; Yang, T.; Sun, J. Anchor detr: Query design for transformer-based detector. In Proceedings of the Association for the Advancement of Artificial Intelligence, Virtual, 22 February–1 March 2022; Volume 36, pp. 2567–2575.
64. Li, F.; Zhang, H.; Liu, S.; Guo, J.; Ni, L.M.; Zhang, L. Dn-detr: Accelerate detr training by introducing query denoising. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 13619–13627.
65. Li, F.; Zhang, H.; Xu, H.; Liu, S.; Zhang, L.; Ni, L.M.; Shum, H.Y. Mask dino: Towards a unified transformer-based framework for object detection and segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 3041–3050.
66. Jaderberg, M.; Simonyan, K.; Vedaldi, A.; Zisserman, A. Reading text in the wild with convolutional neural networks. *Int. J. Comput. Vis.* **2016**, *116*, 1–20. [CrossRef]
67. Liu, W.; Chen, C.; Wong, K.Y. Char-net: A character-aware neural network for distorted scene text recognition. In Proceedings of the Association for the Advancement of Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
68. Cheng, Z.; Xu, Y.; Bai, F.; Niu, Y.; Pu, S.; Zhou, S. Aon: Towards arbitrarily-oriented text recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5571–5579.
69. Li, H.; Wang, P.; Shen, C.; Zhang, G. Show, attend and read: A simple and strong baseline for irregular text recognition. In Proceedings of the Association for the Advancement of Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 8610–8617.
70. Liu, Y.; Wang, Z.; Jin, H.; Wassell, I. Synthetically supervised feature learning for scene text recognition. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 435–451.
71. Chen, T.; Wang, Z.; Xu, N.; Jin, H.; Luo, J. Large-scale tag-based font retrieval with generative feature learning. In Proceedings of the International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9116–9125.
72. Zhu, Y.; Tan, T.; Wang, Y. Font recognition based on global texture analysis. *IEEE Tran. Pattern Anal. Mach. Intell.* **2001**, *23*, 1192–1200.
73. Chen, G.; Yang, J.; Jin, H.; Brandt, J.; Shechtman, E. Large-scale visual font recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 3598–3605.
74. Wang, Z.; Yang, J.; Jin, H.; Shechtman, E.; Agarwala, A.; Brandt, J.; Huang, T.S. Deepfont: Identify your font from an image. In Proceedings of the ACM Multimedia, Brisbane, Australia, 26–30 October 2015; pp. 451–459.
75. Bharath, V.; Rani, N.S. A font style classification system for English OCR. In Proceedings of the International Conference on Intelligent Computing and Control, Coimbatore, India, 23–24 June 2017; pp. 1–5.
76. Liu, Y.; Wang, Z.; Jin, H.; Wassell, I. Multi-task adversarial network for disentangled feature learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 3743–3751.
77. Sun, Q.; Cui, J.; Gu, Z. Extending CLIP for Text-to-font Retrieval. In Proceedings of the International Conference on Multimedia Retrieval, Phuket, Thailand, 10–14 June 2024; pp. 1170–1174.
78. Stoffel, A.; Spretke, D.; Kinnemann, H.; Keim, D.A. Enhancing document structure analysis using visual analytics. In Proceedings of the ACM Symposium on Applied Computing, Sierre, Switzerland, 22–26 March 2010; pp. 8–12.
79. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
80. Patil, A.G.; Ben-Eliezer, O.; Perel, O.; Averbuch-Elor, H. Read: Recursive autoencoders for document layout generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 14–19 June 2020; pp. 544–545.
81. Sun, H.M. Page segmentation for Manhattan and non-Manhattan layout documents via selective CRLA. In Proceedings of the Document Analysis and Recognition, Seoul, Republic of Korea, 31 August–1 September 2005; pp. 116–120.
82. Agrawal, M.; Doermann, D. Voronoi++: A dynamic page segmentation approach based on voronoi and docstrum features. In Proceedings of the Document Analysis and Recognition, Barcelona, Spain, 26–29 July 2009; pp. 1011–1015.
83. Simon, A.; Pret, J.C.; Johnson, A.P. A fast algorithm for bottom-up document layout analysis. *IEEE Tran. Pattern Anal. Mach. Intell.* **1997**, *19*, 273–277. [CrossRef]
84. Tran, T.A.; Na, I.S.; Kim, S.H. Page segmentation using minimum homogeneity algorithm and adaptive mathematical morphology. *Int. J. Doc. Anal. Recognit.* **2016**, *19*, 191–209. [CrossRef]
85. Vil'kin, A.M.; Safonov, I.V.; Egorova, M.A. Algorithm for segmentation of documents based on texture features. *Pattern Recognit. Image Anal.* **2013**, *23*, 153–159. [CrossRef]
86. Grüning, T.; Leifert, G.; Strauß, T.; Michael, J.; Labahn, R. A two-stage method for text line detection in historical documents. *J. Doc. Anal. Recognit.* **2019**, *22*, 285–302. [CrossRef]

87. Xu, Y.; Yin, F.; Zhang, Z.; Liu, C.L. Multi-task Layout Analysis for Historical Handwritten Documents Using Fully Convolutional Networks. In Proceedings of the International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 13–19 July 2018; pp. 1057–1063.
88. Luo, S.; Ivison, H.; Han, S.C.; Poon, J. Local interpretations for explainable natural language processing: A survey. *ACM Comput. Surv.* **2024**, *56*, 1–36. [CrossRef]
89. Kong, W.; Jiang, Z.; Sun, S.; Guo, Z.; Cui, W.; Liu, T.; Lou, J.; Zhang, D. Aesthetics++: Refining graphic designs by exploring design principles and human preference. *IEEE Trans. Vis. Comput. Graph.* **2022**, *29*, 3093–3104. [CrossRef] [PubMed]
90. Son, K.; Oh, S.Y.; Kim, Y.; Choi, H.; Bae, S.H.; Hwang, G. Color sommelier: Interactive color recommendation system based on community-generated color palettes. In Proceedings of the ACM Symposium on User Interface Software & Technology, Charlotte, NC, USA, 8–11 November 2015; pp. 95–96.
91. Jahanian, A.; Liu, J.; Lin, Q.; Tretter, D.; O'Brien-Strain, E.; Lee, S.C.; Lyons, N.; Allebach, J. Recommendation system for automatic design of magazine covers. In Proceedings of the Conference on Intelligent User Interfaces, Santa Monica, CA, USA, 19–22 March 2013; pp. 95–106.
92. Yang, X.; Mei, T.; Xu, Y.Q.; Rui, Y.; Li, S. Automatic generation of visual-textual presentation layout. *ACM Trans. Multimed. Comput. Commun. Appl.* **2016**, *9*, 39. [CrossRef]
93. Maheshwari, P.; Jain, N.; Vaddamanu, P.; Raut, D.; Vaishay, S.; Vinay, V. Generating Compositional Color Representations from Text. In Proceedings of the Conference on Information & Knowledge Management, Gold Coast, QLD, Australia, 1–5 November 2021; pp. 1222–1231.
94. Bahng, H.; Yoo, S.; Cho, W.; Park, D.K.; Wu, Z.; Ma, X.; Choo, J. Coloring with words: Guiding colorization via text-based palette generation. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 431–447.
95. Lu, K.; Feng, M.; Chen, X.; Sedlmair, M.; Deussen, O.; Lischinski, D.; Cheng, Z.; Wang, Y. Palettailor: Discriminable colorization for categorical data. *IEEE Trans. Vis. Comput. Graph.* **2020**, *27*, 475–484. [CrossRef] [PubMed]
96. Yuan, L.P.; Zhou, Z.; Zhao, J.; Guo, Y.; Du, F.; Qu, H. Infocolorizer: Interactive recommendation of color palettes for infographics. *IEEE Trans. Vis. Comput. Graph.* **2021**, *28*, 4252–4266. [CrossRef]
97. Qiu, Q.; Otani, M.; Iwazaki, Y. An intelligent color recommendation tool for landing page design. In Proceedings of the Conference on Intelligent User Interfaces, Helsinki, Finland, 22–25 March 2022; pp. 26–29.
98. Qiu, Q.; Wang, X.; Otani, M.; Iwazaki, Y. Color recommendation for vector graphic documents on multi-palette representation. In Proceedings of the Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–7 January 2023; pp. 3621–3629.
99. Kikuchi, K.; Inoue, N.; Otani, M.; Simo-Serra, E.; Yamaguchi, K. Generative colorization of structured mobile web pages. In Proceedings of the Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–7 January 2023; pp. 3650–3659.
100. Ke, Y.; Tang, X.; Jing, F. The design of high-level features for photo quality assessment. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–17 June 2019; Volume 1, pp. 419–426.
101. Wong, L.K.; Low, K.L. Saliency-enhanced image aesthetics class prediction. In Proceedings of the Conference on Image Processing, Cairo, Egypt, 7–10 November 2009; pp. 997–1000.
102. Dhar, S.; Ordonez, V.; Berg, T.L. High level describable attributes for predicting aesthetics and interestingness. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, 20–25 June 2011; pp. 1657–1664.
103. Obrador, P.; Saad, M.A.; Suryanarayan, P.; Oliver, N. Towards category-based aesthetic models of photographs. In Proceedings of the ACM Multimedia, Nara, Japan, 29 October–2 November 2012; pp. 63–76.
104. Reinecke, K.; Yeh, T.; Miratrix, L.; Mardiko, R.; Zhao, Y.; Liu, J.; Gajos, K.Z. Predicting users' first impressions of website aesthetics with a quantification of perceived visual complexity and colorfulness. In Proceedings of the Conference on Human Factors in Computing Systems, Paris, France, 27 April–2 May 2013; pp. 2049–2058.
105. Lu, X.; Lin, Z.; Jin, H.; Yang, J.; Wang, J.Z. Rapid: Rating pictorial aesthetics using deep learning. In Proceedings of the ACM Multimedia, Orlando, FL, USA, 3–7 November 2014; pp. 457–466.
106. Lu, X.; Lin, Z.; Shen, X.; Mech, R.; Wang, J.Z. Deep multi-patch aggregation network for image aesthetics, and quality estimation. In Proceedings of the International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 990–998.
107. Cui, C.; Lin, P.; Nie, X.; Jian, M.; Yin, Y. Social-sensed image aesthetics assessment. *ACM Trans. Multimed. Comput. Commun. Appl.* **2020**, *16*, 1–19. [CrossRef]
108. Cui, C.; Yang, W.; Shi, C.; Wang, M.; Nie, X.; Yin, Y. Personalized image quality assessment with social-sensed aesthetic preference. *Inf. Sci.* **2020**, *512*, 780–794. [CrossRef]
109. Chen, Q.; Zhang, W.; Zhou, N.; Lei, P.; Xu, Y.; Zheng, Y.; Fan, J. Adaptive fractional dilated convolution network for image aesthetics assessment. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020.

110. Carlier, A.; Danelljan, M.; Alahi, A.; Timofte, R. DeepSVG: A Hierarchical Generative Network for Vector Graphics Animation. In Proceedings of the Advances in Neural Information Processing Systems, Virtual, 6–12 December 2020; Volume 33, pp. 16351–16361.
111. Ha, D.; Eck, D. A Neural Representation of Sketch Drawings. In Proceedings of the International Conference on Learning Representations, Toulon, France, 24–26 April 2017.
112. Lopes, R.G.; Ha, D.; Eck, D.; Shlens, J. A learned representation for scalable vector graphics. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 7930–7939.
113. Wang, Y.; Lian, Z. Deepvecfont: Synthesizing high-quality vector fonts via dual-modality learning. *ACM Trans. Graph. (TOG)* **2021**, *40*, 1–15. [CrossRef]
114. Wu, R.; Su, W.; Ma, K.; Liao, J. IconShop: A Comprehensive Tool for Icon Design and Management. In Proceedings of the International Conference on Computer Graphics and Interactive Techniques, SIGGRAPH, Los Angeles, CA, USA, 6–10 August 2023; pp. 456–465.
115. Das, A.; Yang, Y.; Hospedales, T.; Xiang, T.; Song, Y.Z. Béziersketch: A generative model for scalable vector sketches. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 632–647.
116. Li, T.M.; Lukáč, M.; Gharbi, M.; Ragan-Kelley, J. Differentiable vector graphics rasterization for editing and learning. *ACM Trans. Graph. (TOG)* **2020**, *39*, 1–15. [CrossRef]
117. Ma, X.; Zhou, Y.; Xu, X.; Sun, B.; Filev, V.; Orlov, N.; Fu, Y.; Shi, H. Towards layer-wise image vectorization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–24 June 2022; pp. 16314–16323.
118. Reddy, P.; Gharbi, M.; Lukac, M.; Mitra, N.J. Im2vec: Synthesizing vector graphics without vector supervision. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 19–25 June 2021; pp. 7342–7351.
119. Shen, I.C.; Chen, B.Y. Clipgen: A deep generative model for clipart vectorization and synthesis. *IEEE Trans. Vis. Comput. Graph.* **2021**, *28*, 4211–4224. [CrossRef] [PubMed]
120. Song, Y.; Shao, X.; Chen, K.; Zhang, W.; Jing, Z.; Li, M. Clipvg: Text-guided image manipulation using differentiable vector graphics. In Proceedings of the Association for the Advancement of Artificial Intelligence, Washington, DC, USA, 7–14 February 2023; Volume 37, pp. 2312–2320.
121. Su, H.; Liu, X.; Niu, J.; Cui, J.; Wan, J.; Wu, X.; Wang, N. Marvel: Raster gray-level manga vectorization via primitive-wise deep reinforcement learning. *Trans. Circuits Syst. Video Technol.* **2023**, *34*, 2677–2693. [CrossRef]
122. Xing, X.; Wang, C.; Zhou, H.; Zhang, J.; Yu, Q.; Xu, D. Diffsketcher: Text guided vector sketch synthesis through latent diffusion models. In Proceedings of the Advances in Neural Information Processing Systems, New Orleans, LA, USA, 10–16 December 2023.
123. Yamaguchi, K. Canvasvae: Learning to generate vector graphic documents. In Proceedings of the International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 5481–5489.
124. Frans, K.; Soros, L.; Witkowski, O. Clipdraw: Exploring text-to-drawing synthesis through language-image encoders. In Proceedings of the Advances in Neural Information Processing Systems, New Orleans, LA, USA, 28 November–9 December 2022.
125. Vinker, Y.; Pajouheshgar, E.; Bo, J.Y.; Bachmann, R.C.; Bermano, A.H.; Cohen-Or, D.; Zamir, A.; Shamir, A. Clipasso: Semantically-aware object sketching. *ACM Trans. Graph. (TOG)* **2022**, *41*, 1–11. [CrossRef]
126. Jain, A.; Xie, A.; Abbeel, P. Vectorfusion: Text-to-svg by abstracting pixel-based diffusion models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023.
127. Iluz, S.; Vinker, Y.; Hertz, A.; Berio, D.; Cohen-Or, D.; Shamir, A. Word-as-image for semantic typography. *ACM Trans. Graph. (TOG)* **2023**, *42*, 1–11. [CrossRef]
128. Gal, R.; Vinker, Y.; Alaluf, Y.; Bermano, A.; Cohen-Or, D.; Shamir, A.; Chechik, G. Breathing Life Into Sketches Using Text-to-Video Priors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–22 June 2024; pp. 4325–4336.
129. Xing, X.; Zhou, H.; Wang, C.; Zhang, J.; Xu, D.; Yu, Q. SVGDreamer: Text guided SVG generation with diffusion model. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 17–21 June 2024; pp. 4546–4555.
130. Zhang, P., Zhao, N.; Liao, J. Text-to-Vector Generation with Neural Path Representation. *ACM Trans. Graph. (TOG)* **2024**, *43*, 1–13.
131. Sun, J.; Liang, L.; Wen, F.; Shum, H.Y. Image vectorization using optimized gradient meshes. *ACM Trans. Graph. (TOG)* **2007**, *26*, 11-es. [CrossRef]
132. Xia, T.; Liao, B.; Yu, Y. Patch-based image vectorization with automatic curvilinear feature alignment. *ACM Trans. Graph. (TOG)* **2009**, *28*, 1–10. [CrossRef]
133. Lai, Y.K.; Hu, S.M.; Martin, R.R. Automatic and topology-preserving gradient mesh generation for image vectorization. *ACM Trans. Graph. (TOG)* **2009**, *28*, 1–8. [CrossRef]
134. Zhang, S.H.; Chen, T.; Zhang, Y.F.; Hu, S.M.; Martin, R.R. Vectorizing cartoon animations. *IEEE Trans. Vis. Comput. Graph.* **2009**, *15*, 618–629. [CrossRef] [PubMed]

135. Sỳkora, D.; Buriánek, J.; Zara, J. Sketching Cartoons by Example. In Proceedings of the SBM, Philadelphia, PA, USA, 13–16 March 2005; pp. 27–33.
136. Bessmeltsev, M.; Solomon, J. Vectorization of line drawings via polyvector fields. *ACM Trans. Graph. (TOG)* **2019**, *38*, 1–12. [CrossRef]
137. Dominici, E.A.; Schertler, N.; Griffin, J.; Hoshyari, S.; Sigal, L.; Sheffer, A. Polyfit: Perception-aligned vectorization of raster clip-art via intermediate polygonal fitting. *ACM Trans. Graph.* **2020**, *39*, 77:1–77:16. [CrossRef]
138. Dominici, E.A.; Schertler, N.; Griffin, J.; Hoshyari, S.; Sigal, L.; Sheffer, A. SAMVG: A multi-stage image vectorization model with the segment-anything model. In Proceedings of the International Conference on Acoustics, Speech and Signal Processing, Seoul, Republic of Korea, 14–19 April 2024; pp. 4350–4354.
139. Chen, Y.; Ni, B.; Chen, X.; Hu, Z. Editable image geometric abstraction via neural primitive assembly. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 23514–23523
140. Hu, T.; Yi, R.; Qian, B.; Zhang, J.; Rosin, P.L.; Lai, Y.K. SuperSVG: Superpixel-based scalable vector graphics synthesis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 17–21 June 2024; pp. 24892–24901.
141. Zhou, B.; Wang, W.; Chen, Z. Easy generation of personal Chinese handwritten fonts. In Proceedings of the IEEE International Conference on Multimedia & Expo, Barcelona, Spain, 11–15 July 2011; pp. 1–6.
142. Lian, Z.; Zhao, B.; Xiao, J. Automatic generation of large-scale handwriting fonts via style. In Proceedings of the SIGGRAPH Asia, Macao SAR, China, 5–8 December 2016; pp. 1–4.
143. Phan, H.Q.; Fu, H.; Chan, A.B. Flexyfont: Learning transferring rules for flexible typeface synthesis. In Proceedings of the Computer Graphics Forum, Beijing, China, 7–9 October 2015; Volume 34, pp. 245–256.
144. Tenenbaum, J.; Freeman, W. Separating style and content. In Proceedings of the Advances in Neural Information Processing Systems, Denver, CO, USA, 2–5 December 1996; Volume 9.
145. Goda, Y.; Nakamura, T.; Kanoh, M. Texture transfer based on continuous structure of texture patches for design of artistic Shodo fonts. In Proceedings of the ACM SIGGRAPH ASIA, Seoul, Republic of Korea, 15–18 December 2010; pp. 1–2.
146. Murata, K.; Nakamura, T.; Endo, K.; Kanoh, M.; Yamada, K. Japanese Kanji-calligraphic font design using onomatopoeia utterance. In Proceedings of the Congress on Evolutionary Computation, Vancouver, BC, Canada, 24–29 July 2016; pp. 1708–1713.
147. Tian, Y. zi2zi: Master chinese calligraphy with conditional adversarial networks. *Internet* **2017**, *3*, 2.
148. Jiang, Y.; Lian, Z.; Tang, Y.; Xiao, J. Dcfont: An end-to-end deep chinese font generation system. In Proceedings of the SIGGRAPH Asia, Bangkok, Thailand, 27–30 November 2017; pp. 1–4.
149. Huang, Y.; He, M.; Jin, L.; Wang, Y. Rd-gan: Few/zero-shot chinese character style transfer via radical decomposition and rendering. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 156–172.
150. Zhang, Y.; Zhang, Y.; Cai, W. Separating style and content for generalized style transfer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 847–855.
151. Gao, Y.; Guo, Y.; Lian, Z.; Tang, Y.; Xiao, J. Artistic glyph image synthesis via one-stage few-shot learning. *ACM Trans. Graph.* **2019**, *38*, 185. [CrossRef]
152. Xie, Y.; Chen, X.; Sun, L.; Lu, Y. Dg-font: Deformable generative networks for unsupervised font generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 19–25 June 2021.
153. Yang, S.; Liu, J.; Lian, Z.; Guo, Z. Awesome typography: Statistics-based text effects transfer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7464–7473.
154. Men, Y.; Lian, Z.; Tang, Y.; Xiao, J. A common framework for interactive texture transfer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6353–6362.
155. Azadi, S.; Fisher, M.; Kim, V.G.; Wang, Z.; Shechtman, E.; Darrell, T. Multi-content gan for few-shot font style transfer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7564–7573.
156. Chen, F.; Wang, Y.; Xu, S.; Wang, F.; Sun, F.; Jia, X. Style transfer network for complex multi-stroke text. *Multimed. Syst.* **2023**, *9*, 91–100. [CrossRef]
157. Xue, M.; Ito, Y.; Nakano, K. An Art Font Generation Technique using Pix2Pix-based Networks. *Bull. Netw. Comput. Syst. Softw.* **2023**, *12*, 6–12.
158. Wang, C.; Wu, L.; Liu, X.; Li, X.; Meng, L.; Meng, X. Anything to glyph: Artistic font synthesis via text-to-image diffusion model. In Proceedings of the SIGGRAPH Asia, Sydney, Australia, 12–15 December 2023; pp. 1–11.
159. Xu, J.; Kaplan, C.S. Calligraphic packing. In Proceedings of the Graphics Interface, Montreal, QC, Canada, 27–29 May 2007; pp. 43–50.
160. Zou, C.; Cao, J.; Ranaweera, W.; Alhashim, I.; Tan, P.; Sheffer, A.; Zhang, H. Legible compact calligrams. *ACM Trans. Graph. (TOG)* **2016**, *35*, 1–12. [CrossRef]

161. Zhang, J.; Wang, Y.; Xiao, W.; Luo, Z. Synthesizing ornamental typefaces. In Proceedings of the Computer Graphics Forum, Yokohama, Japan, 12–16 June 2017; Volume 36, pp. 64–75.
162. Tanveer, M.; Wang, Y.; Mahdavi-Amiri, A.; Zhang, H. Ds-fusion: Artistic typography via discriminated and stylized diffusion. In Proceedings of the International Conference on Computer Vision, Paris, France, 2–6 October 2023; pp. 374–384.
163. Gupta, A.; Vedaldi, A.; Zisserman, A. Synthetic data for text localisation in natural images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2315–2324.
164. Zhan, F.; Lu, S.; Xue, C. Verisimilar image synthesis for accurate detection and recognition of texts. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 249–266.
165. Liao, M.; Song, B.; Long, S.; He, M.; Yao, C.; Bai, X. SynthText3D: Synthesizing scene text images from 3D virtual worlds. *Sci. China Inf. Sci.* **2020**, *63*, 120105. [CrossRef]
166. Chen, J.; Huang, Y.; Lv, T.; Cui, L.; Chen, Q.; Wei, F. Textdiffuser-2: Unleashing the power of language models for text rendering. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 386–402.
167. Tuo, Y.; Xiang, W.; He, J.Y.; Geng, Y.; Xie, X. Anytext: Multilingual visual text generation and editing. *arXiv* **2023**, arXiv:2311.03054.
168. Esser, P.; Kulal, S.; Blattmann, A.; Entezari, R.; Müller, J.; Saini, H.; Levi, Y.; Lorenz, D.; Sauer, A.; Boesel, F.; et al. Scaling rectified flow transformers for high-resolution image synthesis. In Proceedings of the International Conference on Machine Learning, Vienna, Austria, 21–27 July 2024.
169. Saharia, C.; Chan, W.; Saxena, S.; Li, L.; Whang, J.; Denton, E.L.; Ghasemipour, K.; Gontijo Lopes, R.; Karagol Ayan, B.; Salimans, T.; et al. Photorealistic text-to-image diffusion models with deep language understanding. In Proceedings of the Advances in Neural Information Processing Systems, New Orleans, LA, USA, 28 November–9 December 2022; Volume 35, pp. 36479–36494.
170. Xue, L.; Barua, A.; Constant, N.; Al-Rfou, R.; Narang, S.; Kale, M.; Roberts, A.; Raffel, C. Byt5: Towards a token-free future with pre-trained byte-to-byte models. *Trans. Assoc. Comput. Linguist.* **2022**, *10*, 291–306. [CrossRef]
171. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning transferable visual models from natural language supervision. In Proceedings of the International Conference on Machine Learning, Virtual, 18–24 July 2021; pp. 8748–8763.
172. Ma, J.; Zhao, M.; Chen, C.; Wang, R.; Niu, D.; Lu, H.; Lin, X. GlyphDraw: Seamlessly Rendering Text with Intricate Spatial Structures in Text-to-Image Generation. *arXiv* **2023**, arXiv:2303.17870.
173. Yang, Y.; Gui, D.; Yuan, Y.; Liang, W.; Ding, H.; Hu, H.; Chen, K. Glyphcontrol: Glyph conditional control for visual text generation. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 9–15 December 2024; Volume 36, pp. 44050–44066.
174. Zhang, L.; Chen, X.; Wang, Y.; Lu, Y.; Qiao, Y. Brush your text: Synthesize any scene text on images via diffusion model. In Proceedings of the Association for the Advancement of Artificial Intelligence, Vancouver, BC, Canada, 20–27 February 2024; Volume 38, pp. 7215–7223.
175. Li, Z.; Shu, Y.; Zeng, W.; Yang, D.; Zhou, Y. First Creating Backgrounds Then Rendering Texts: A New Paradigm for Visual Text Blending. *arXiv* **2024**, arXiv:2410.10168. [CrossRef]
176. Lee, B.; Srivastava, S.; Kumar, R.; Brafman, R.; Klemmer, S.R. Designing with interactive example galleries. In Proceedings of the Conference on Human Factors in Computing Systems, Atlanta, GA, USA, 10–15 April 2010; pp. 2257–2266.
177. Dayama, N.R.; Todi, K.; Saarelainen, T.; Oulasvirta, A. Grids: Interactive layout design with integer programming. In Proceedings of the Conference on Human Factors in Computing Systems, Honolulu, HI, USA, 25–30 April 2020; pp. 1–13.
178. O'Donovan, P.; Agarwala, A.; Hertzmann, A. Designscape: Design with interactive layout suggestions. In Proceedings of the Conference on Human Factors in Computing Systems, Honolulu, HI, USA, 25–30 April 2015; pp. 1221–1224.
179. Todi, K.; Weir, D.; Oulasvirta, A. Sketchplore: Sketch and explore with a layout optimiser. In Proceedings of the Conference on designing interactive systems, Brisbane, Australia, 4–8 June 2016; pp. 543–555.
180. Lee, H.Y.; Jiang, L.; Essa, I.; Le, P.B.; Gong, H.; Yang, M.H.; Yang, W. Neural design network: Graphic layout generation with constraints. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 491–506.
181. Li, J.; Yang, J.; Zhang, J.; Liu, C.; Wang, C.; Xu, T. Attribute-conditioned layout gan for automatic graphic design. *IEEE Trans. Vis. Comput. Graph.* **2020**, *27*, 4039–4048. [CrossRef]
182. Jing, Q.; Zhou, T.; Tsang, Y.; Chen, L.; Sun, L.; Zhen, Y.; Du, Y. Layout generation for various scenarios in mobile shopping applications. In Proceedings of the Conference on Human Factors in Computing Systems, Hamburg, Germany, 23–28 April 2023; pp. 1–18.
183. Zhang, W.; Zheng, Y.; Miyazono, T.; Uchida, S.; Iwana, B.K. Towards book cover design via layout graphs. In *Document Analysis and Recognition*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 642–657.
184. Chai, S.; Zhuang, L.; Yan, F.; Zhou, Z. Two-stage Content-Aware Layout Generation for Poster Designs. In Proceedings of the ACM Multimedia, Vancouver, BC, Canada, 7–10 June 2023; pp. 8415–8423.
185. Guo, S.; Jin, Z.; Sun, F.; Li, J.; Li, Z.; Shi, Y.; Cao, N. Vinci: An intelligent graphic design system for generating advertising posters. In Proceedings of the Conference on Human Factors in Computing Systems, Virtual, 8–13 May 2021; pp. 1–17.

186. Damera-Venkata, N.; Bento, J.; O'Brien-Strain, E. Probabilistic document model for automated document composition. In Proceedings of the ACM Symposium on Document Engineering, Mountain View, CA, USA, 19–22 September 2011; pp. 3–12.
187. Hurst, N.; Li, W.; Marriott, K. Review of automatic document formatting. In Proceedings of the Symposium on Document Engineering, Munich, Germany, 15–18 September 2009; pp. 99–108.
188. O'Donovan, P.; Agarwala, A.; Hertzmann, A. Learning layouts for single-pagegraphic designs. *IEEE Trans. Vis. Comput. Graph.* **2014**, *20*, 1200–1213. [CrossRef]
189. Tabata, S.; Yoshihara, H.; Maeda, H.; Yokoyama, K. Automatic layout generation for graphical design magazines. In Proceedings of the ACM SIGGRAPH, Los Angeles, CA, USA, 28 July 2019; pp. 1–2.
190. Bylinskii, Z.; Kim, N.W.; O'Donovan, P.; Alsheikh, S.; Madan, S.; Pfister, H.; Durand, F.; Russell, B.; Hertzmann, A. Learning visual importance for graphic designs and data visualizations. In Proceedings of the ACM Symposium on User Interface Software and Technology, Québec City, QC, Canada, 22–25 October 2017; pp. 57–69.
191. Pang, X.; Cao, Y.; Lau, R.W.; Chan, A.B. Directing user attention via visual flow on web designs. *ACM Trans. Graph.* **2016**, *35*, 1–11. [CrossRef]
192. Kikuchi, K.; Simo-Serra, E.; Otani, M.; Yamaguchi, K. Constrained graphic layout generation via latent optimization. In Proceedings of the ACM Multimedia, Vitural, 20–24 October 2021; pp. 88–96.
193. Chai, S.; Zhuang, L.; Yan, F. Layoutdm: Transformer-based diffusion model for layout generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023.
194. Zhang, J.; Guo, J.; Sun, S.; Lou, J.G.; Zhang, D. Layoutdiffusion: Improving graphic layout generation by discrete diffusion probabilistic models. In Proceedings of the International Conference on Computer Vision, Paris, France, 2–6 October 2023; pp. 7226–7236.
195. Hui, M.; Zhang, Z.; Zhang, X.; Xie, W.; Wang, Y.; Lu, Y. Unifying layout generation with a decoupled diffusion model. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 1942–1951.
196. Shabani, M.A.; Wang, Z.; Liu, D.; Zhao, N.; Yang, J.; Furukawa, Y. Visual Layout Composer: Image-Vector Dual Diffusion Model for Design Layout Generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 17–21 June 2024; pp. 9222–9231.
197. Jyothi, A.A.; Durand, T.; He, J.; Sigal, L.; Mori, G. Layoutvae: Stochastic scene layout generation from a label set. In Proceedings of the International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9895–9904.
198. Gupta, K.; Lazarow, J.; Achille, A.; Davis, L.S.; Mahadevan, V.; Shrivastava, A. Layouttransformer: Layout generation and completion with self-attention. In Proceedings of the International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 1004–1014.
199. Arroyo, D.M.; Postels, J.; Tombari, F. Variational transformer networks for layout generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, 20–25 June 2011; pp. 13642–13652.
200. Kong, X.; Jiang, L.; Chang, H.; Zhang, H.; Hao, Y.; Gong, H.; Essa, I. Blt: Bidirectional layout transformer for controllable layout generation. In Proceedings of the European Conference on Computer Vision, Tel-Aviv, Israel, 23–27 October 2022; pp. 474–490.
201. Inoue, N.; Kikuchi, K.; Simo-Serra, E.; Otani, M.; Yamaguchi, K. Layoutdm: Discrete diffusion model for controllable layout generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023.
202. Zhang, Z.; Zhang, Y.; Liang, Y.; Xiang, L.; Zhao, Y.; Zhou, Y.; Zong, C. Layoutdit: Layout-aware end-to-end document image translation with multi-step conductive decoder. In Proceedings of the Empirical Methods in Natural Language Processing, Singapore, 6–10 December 2023; pp. 10043–10053.
203. Levi, E.; Brosh, E.; Mykhailych, M.; Perez, M. Dlt: Conditioned layout generation with joint discrete-continuous diffusion layout transformer. In Proceedings of the International Conference on Computer Vision, Paris, France, 2–6 October 2023; pp. 2106–2115.
204. Zheng, X.; Qiao, X.; Cao, Y.; Lau, R.W. Content-aware generative modeling of graphic design layouts. *ACM Trans. Graph.* **2019**, *38*, 1–15. [CrossRef]
205. Cao, Y.; Ma, Y.; Zhou, M.; Liu, C.; Xie, H.; Ge, T.; Jiang, Y. Geometry aligned variational transformer for image-conditioned layout generation. In Proceedings of the ACM Multimedia, Lisbon, Portugal, 10–14 October 2022; pp. 1561–1571.
206. Hsu, H.Y.; He, X.; Peng, Y.; Kong, H.; Zhang, Q. Posterlayout: A new benchmark and approach for content-aware visual-textual presentation layout. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 6018–6026.
207. Yu, N.; Chen, C.C.; Chen, Z.; Meng, R.; Wu, G.; Josel, P.; Niebles, J.C.; Xiong, C.; Xu, R. LayoutDETR: Detection transformer is a good multimodal layout designer. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 October 2025; pp. 169–187.
208. Tang, Z.; Wu, C.; Li, J.; Duan, N. Layoutnuwa: Revealing the hidden layout expertise of large language models. *arXiv* **2023**, arXiv:2309.09506. [CrossRef]

209. Li, S.; Wang, R.; Hsieh, C.J.; Cheng, M.; Zhou, T. MuLan: Multimodal-LLM Agent for Progressive Multi-Object Diffusion. *arXiv* **2024**, arXiv:2402.12741.
210. Chen, J.; Zhang, R.; Zhou, Y.; Healey, J.; Gu, J.; Xu, Z.; Chen, C. TextLap: Customizing Language Models for Text-to-Layout Planning. *arXiv* **2024**, arXiv:2410.12844.
211. Lin, J.; Guo, J.; Sun, S.; Yang, Z.; Lou, J.G.; Zhang, D. Layoutprompter: Awaken the design ability of large language models. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 9–15 December 2024; Volume 36, pp. 43852–43879.
212. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. In Proceedings of the Advances in Neural Information Processing Systems, Vitural, 6–12 December 2020; Volume 33, pp. 1877–1901.
213. Wang, Y.; Pu, G.; Luo, W.; Wang, Y.; Xiong, P.; Kang, H.; Lian, Z. Aesthetic text logo synthesis via content-aware layout inferring. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 2436–2445.
214. He, J.; Wang, Y.; Wang, L.; Lu, H.; He, J.Y.; Li, C.; Chen, H.; Lan, J.P.; Luo, B.; Geng, Y. GLDesigner: Leveraging Multi-Modal LLMs as Designer for Enhanced Aesthetic Text Glyph Layouts. *arXiv* **2024**, arXiv:2411.11435. [CrossRef]
215. Lakhanpal, S.; Chopra, S.; Jain, V.; Chadha, A.; Luo, M. Refining Text-to-Image Generation: Towards Accurate Training-Free Glyph-Enhanced Image Generation. *arXiv* **2024**, arXiv:2403.16422.
216. Ouyang, D.; Furuta, R.; Shimizu, Y.; Taniguchi, Y.; Hinami, R.; Ishiwatari, S. Interactive manga colorization with fast flat coloring. In Proceedings of the SIGGRAPH Asia, Tokyo, Japan, 14–17 December 2021; pp. 1–2.
217. Yuan, M.; Simo-Serra, E. Line art colorization with concatenated spatial attention. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vitural, 19–25 June 2021; pp. 3946–3950.
218. Zhang, Q.; Wang, B.; Wen, W.; Li, H.; Liu, J. Line art correlation matching feature transfer network for automatic animation colorization. In Proceedings of the Winter Conference on Applications of Computer Vision, Virtual, 5–9 June 2021; pp. 3872–3881.
219. Wang, N.; Niu, M.; Wang, Z.; Hu, K.; Liu, B.; Wang, Z.; Li, H. Region assisted sketch colorization. *IEEE Trans. Image Process.* **2023**, *32*, 6142–6154. [CrossRef]
220. Zabari, N.; Azulay, A.; Gorkor, A.; Halperin, T.; Fried, O. Diffusing Colors: Image Colorization with Text Guided Diffusion. In Proceedings of the SIGGRAPH Asia, Sydney, Australia, 12–15 December 2023; pp. 1–11.
221. Dou, Z.; Wang, N.; Li, B.; Wang, Z.; Li, H.; Liu, B. Dual color space guided sketch colorization. *IEEE Trans. Image Process.* **2021**, *30*, 7292–7304. [CrossRef]
222. Yun, J.; Lee, S.; Park, M.; Choo, J. iColoriT: Towards propagating local hints to the right region in interactive colorization by leveraging vision transformer. In Proceedings of the Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 2–7 January 2023; pp. 1787–1796.
223. Zhang, L.; Li, C.; Simo-Serra, E.; Ji, Y.; Wong, T.T.; Liu, C. User-guided line art flat filling with split filling mechanism. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vitural, 19–25 June 2021; pp. 9889–9898.
224. Bai, Y.; Dong, C.; Chai, Z.; Wang, A.; Xu, Z.; Yuan, C. Semantic-sparse colorization network for deep exemplar-based colorization. In Proceedings of the European Conference on Computer Vision, Tel-Aviv, Israel, 23–27 October 2022.
225. Li, H.; Sheng, B.; Li, P.; Ali, R.; Chen, C.P. Globally and locally semantic colorization via exemplar-based broad-GAN. *IEEE Trans. Image Process.* **2021**, *30*, 8526–8539. [CrossRef]
226. Li, Y.K.; Lien, Y.H.; Wang, Y.S. Style-structure disentangled features and normalizing flows for icon colorization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022.
227. Li, Z.; Geng, Z.; Kang, Z.; Chen, W.; Yang, Y. Eliminating gradient conflict in reference-based line-art colorization. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; pp. 579–596.
228. Wang, H.; Zhai, D.; Liu, X.; Jiang, J.; Gao, W. Unsupervised deep exemplar colorization via pyramid dual non-local attention. *IEEE Trans. Image Process.* **2023**, *32*, 4114–4127. [CrossRef]
229. Wu, S.; Yan, X.; Liu, W.; Xu, S.; Zhang, S. Self-driven dual-path learning for reference-based line art colorization under limited data. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *34*, 1388–1402. [CrossRef]
230. Wu, S.; Yang, Y.; Xu, S.; Liu, W.; Yan, X.; Zhang, S. Flexicon: Flexible icon colorization via guided images and palettes. In Proceedings of the ACM Multimedia, Vancouver, BC, Canada, 7–10 June 2023; pp. 8662–8673.
231. Zhang, J.; Xu, C.; Li, J.; Han, Y.; Wang, Y.; Tai, Y.; Liu, Y. Scsnet: An efficient paradigm for learning simultaneously image colorization and super-resolution. In Proceedings of the Association for the Advancement of Artificial Intelligence, Virtual, 22 February–1 March 2022; Volume 36, pp. 3271–3279.
232. Zou, C.; Wan, S.; Blanch, M.G.; Murn, L.; Mrak, M.; Sock, J.; Yang, F.; Herranz, L. Lightweight Exemplar Colorization via Semantic Attention-Guided Laplacian Pyramid. *IEEE Trans. Vis. Comput. Graph.* **2024**, *31*, 4257–4269. [CrossRef]
233. Wang, Y.; Xia, M.; Qi, L.; Shao, J.; Qiao, Y. PalGAN: Image colorization with palette generative adversarial networks. In Proceedings of the European Conference on Computer Vision, Tel-Aviv, Israel, 23–27 October 2022; pp. 271–288.

234. Chang, Z.; Weng, S.; Zhang, P.; Li, Y.; Li, S.; Shi, B. L-CoIns: Language-based colorization with instance awareness. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023.
235. Weng, S.; Zhang, P.; Li, Y.; Li, S.; Shi, B. L-cad: Language-based colorization with any-level descriptions using diffusion priors. In Proceedings of the Advances in Neural Information Processing Systems, New Orleans, LA, USA, 10–16 December 2023; Volume 36, pp. 77174–77186.
236. Zhang, L.; Rao, A.; Agrawala, M. Adding conditional control to text-to-image diffusion models. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 1–6 October 2023; pp. 3836–3847.
237. Zhang, L.; Li, C.; Wong, T.T.; Ji, Y.; Liu, C. Two-stage sketch colorization. *ACM Trans. Graph. (TOG)* **2018**, *37*, 1–14. [CrossRef]
238. Cheng, Z.; Yang, Q.; Sheng, B. Deep colorization. In Proceedings of the International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 415–423.
239. Endo, Y.; Iizuka, S.; Kanamori, Y.; Mitani, J. Deepprop: Extracting deep features from a single image for edit propagation. In Proceedings of the 37th Annual Conference of the European Association for Computer Graphics, Lisbon, Portugal, 9–13 May 2016; Volume 35, pp. 189–201.
240. Chang, H.; Fried, O.; Liu, Y.; DiVerdi, S.; Finkelstein, A. Palette-based photo recoloring. *ACM Trans. Graph. (TOG)* **2015**, *34*, 139. [CrossRef]
241. Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; Ommer, B. High-resolution image synthesis with latent diffusion models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–24 June 2022; pp. 10684–10695.
242. Tatsukawa, Y.; Shen, I.C.; Qi, A.; Koyama, Y.; Igarashi, T.; Shamir, A. FontCLIP: A Semantic Typography Visual-Language Model for Multilingual Font Applications. *Comput. Graph. Forum* **2024**, *43*, e15043. [CrossRef]
243. Olsen, L.H.; Glad, I.K.; Jullum, M.; Aas, K. Using Shapley values and variational autoencoders to explain predictive models with dependent mixed features. *J. Mach. Learn. Res.* **2022**, *23*, 1–51.
244. Chen, J.; Huang, Y.; Lv, T.; Cui, L.; Chen, Q.; Wei, F. Textdiffuser: Diffusion models as text painters. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 9353–9387.
245. Zhang, J.; Zhou, Y.; Gu, J.; Wigington, C.; Yu, T.; Chen, Y.; Sun, T.; Zhang, R. Artist: Improving the generation of text-rich images with disentangled diffusion models and large language models. In Proceedings of the 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (Winter Conference on Applications of Computer Vision), Tucson, AZ, USA, 26 February–6 March 2025; pp. 1268–1278.
246. Li, J.; Yang, J.; Hertzmann, A.; Zhang, J.; Xu, T. Layoutgan: Synthesizing graphic layouts with vector-wireframe adversarial networks. *IEEE Tran. Pattern Anal. Mach. Intell.* **2020**, *43*, 2388–2399. [CrossRef]
247. Dong, R.; Han, C.; Peng, Y.; Qi, Z.; Ge, Z.; Yang, J.; Zhao, L.; Sun, J.; Zhou, H.; Wei, H.; et al. DreamLLM: Synergistic multimodal comprehension and creation. *arXiv* **2023**, arXiv:2309.11499.
248. Kikuchi, K.; Inoue, N.; Otani, M.; Simo-Serra, E.; Yamaguchi, K. Multimodal Markup Document Models for Graphic Design Completion. *arXiv* **2024**, arXiv:2409.19051. [CrossRef]
249. Inoue, N.; Masui, K.; Shimoda, W.; Yamaguchi, K. OpenCOLE: Towards Reproducible Automatic Graphic Design Generation. *arXiv* **2024**, arXiv:2406.08232. [CrossRef]
250. Alayrac, J.B.; Donahue, J.; Luc, P.; Miech, A.; Barr, I.; Hasson, Y.; Lenc, K.; Mensch, A.; Millican, K.; Reynolds, M.; et al. Flamingo: A visual language model for few-shot learning. In Proceedings of the Advances in Neural Information Processing Systems, New Orleans, LA, USA, 28 November–9 December 2022; Volume 35, pp. 23716–23736.
251. Awadalla, A.; Gao, I.; Gardner, J.; Hessel, J.; Hanafy, Y.; Zhu, W.; Marathe, K.; Bitton, Y.; Gadre, S.; Sagawa, S.; et al. OpenFlamingo: An open-source framework for training large autoregressive vision-language models. *arXiv* **2023**, arXiv:2308.01390.
252. Esser, P.; Rombach, R.; Ommer, B. Taming transformers for high-resolution image synthesis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 12873–12883.
253. Zou, X.; Zhang, W.; Zhao, N. From Fragment to One Piece: A Survey on AI-Driven Graphic Design. *arXiv* **2025**, arXiv:2503.18641. [CrossRef]
254. Qu, L.; Wu, S.; Fei, H.; Nie, L.; Chua, T.S. Layoutllm-t2i: Eliciting layout guidance from llm for text-to-image generation. In Proceedings of the ACM Multimedia, Ottawa, ON, Canada, 29 October–3 November 2023; pp. 643–654.
255. Gao, P.; Geng, S.; Zhang, R.; Ma, T.; Fang, R.; Zhang, Y.; Li, H.; Qiao, Y. Clip-adapter: Better vision-language models with feature adapters. *Int. J. Comput. Vis.* **2024**, *132*, 581–595. [CrossRef]
256. Zhang, J.; Yoshihashi, R.; Kitada, S.; Osanai, A.; Nakashima, Y. VASCAR: Content-Aware Layout Generation via Visual-Aware Self-Correction. *arXiv* **2024**, arXiv:2412.04237v3.
257. Goyal, S.; Mahajan, A.; Mishra, S.; Udhayanan, P.; Shukla, T.; Joseph, K.J.; Srinivasan, B.V. Design-o-meter: Towards Evaluating and Refining Graphic Designs. In Proceedings of the 2025 IEEE/CVF Winter Conference on Applications of Computer Vision, Tucson, AZ, USA, 28 February–4 March 2025; Volume 5, pp. 676–5686.

258. Zhang, K.; Mo, L.; Chen, W.; Sun, H.; Su, Y. MagicBrush: A Multimodal Assistant for Visual Design Editing. In Proceedings of the Advances in Neural Information Processing Systems, New Orleans, LA, USA, 10–16 December 2023; Volume 36, pp. 31428–31449.
259. Nakano, R.; Hilton, J.; Balaji, S.; Wu, J.; Ouyang, L.; Kim, C.; Hesse, C.; Jain, S.; Kosaraju, V.; Saunders, W.; et al. WebGPT: Browser-assisted question-answering with human feedback. *arXiv* **2021**, arXiv:2112.09332.
260. Le D.H.; Pham, T.; Lee, S.; Clark, C.; Kembhavi, A.; Mandt, S.; Krishna, R.; Lu, J. One Diffusion to Generate Them All. In Proceedings of the Computer Vision and Pattern Recognition Conference, Music City Center, Nashville, TN, USA, 11–15 June 2025; pp. 2671–2682.
261. Wang, Z.; Hsu, J.; Wang, X.; Huang, K.H.; Li, M.; Wu, J.; Ji, H. Visually Descriptive Language Model for Vector Graphics Reasoning. *arXiv* **2024**, arXiv:2404.06479.