APSIPA Transactions on Signal and Information Processing, 2024, 13, e21

Overview Paper

Intelligent Artistic Typography: A Comprehensive Review of Artistic Text Design and Generation

Yuhang Bai*, Zichuan Huang*, Wenshuo Gao, Shuai Yang† and Jiaying Liu

Wangxuan Institute of Computer Technology, Peking University

ABSTRACT

Artistic text generation aims to amplify the aesthetic qualities of text while maintaining readability. It can make the text more attractive and better convey its expression, thus enjoying a wide range of application scenarios such as social media display, consumer electronics, fashion, and graphic design. Artistic text generation includes artistic text stylization and semantic typography. Artistic text stylization concentrates on the text effect overlaid upon the text, such as shadows, outlines, colors, glows, and textures. By comparison, semantic typography focuses on the deformation of the characters to strengthen their visual representation by mimicking the semantic understanding within the text. This overview paper provides an introduction to both artistic text stylization and semantic typography, including the taxonomy, the key ideas of representative methods, and the applications in static and dynamic artistic text generation. Furthermore, the dataset and evaluation metrics are introduced, and the future directions of artistic text generation are discussed. A comprehensive list of artistic text generation models studied in this review is available at https://github.com/williamyang1991/Awesome-Artistic-Typography/.

* Equal contributions.
† Corresponding author: Shuai Yang, williamyang@pku.edu.cn

Received 27 June 2024; revised 07 August 2024; accepted 08 August 2024
ISSN 2048-7703; DOI 10.1561/116.20240037

Keywords: Artistic text, text effect, semantic typography, kinetic typography, style transfer, AIGC.

1 Introduction

Artistic text generation focuses on turning text into visual forms that increase their artistic expression or convey their meaning. It can integrate plain text with fantastic style, decoration, and appearance, creating typography that is legible, readable, and appealing. Such integration of visual representation and semantic understanding, not only attracts viewers, but also emphasizes the messages' meaning and strengthens the impact, making artistic text generation prevalent in graphic design, manga and comic book industry, advertisement, websites, consumer electronics, and social media.

Artistic text generation can be broadly classified into two categories: artistic text stylization and semantic typography. The former primarily involves applying visual effects (*i.e.*, text effects) to text, while the latter focuses on redesigning the shape of text to match specific objects, as illustrated in Figure 1. Furthermore, in the era of mobile internet and multimedia, incorporating motion into artistic text to create dynamic artistic text has gained significant attention due to its captivating nature. However, the manual creation of such typographical art poses considerable challenges: it demands substantial time and effort. Consequently, with the advancement of computer technology, computer-assisted and even fully automated approaches for artistic text rendering and design have emerged.

Artistic text rendering pertains to artistic image rendering, a powerful tool for the automated generation of artistic images. The field of artistic image rendering has a long research history, starting from early methods based on traditional stroke-based [56, 29, 28] and patch-based [30, 14, 17] approaches, to the era of deep learning with techniques like Neural Style Transfer [20, 42] and Generative Adversarial Networks (GANs) [22, 40, 134]. In the current era of AI-Generated Content (AIGC), empowered by powerful large-scale models [31, 76], we are now capable of generating highly fascinating artistic images. However, the text differs significantly from natural images or real artwork. First, text is highly abstract and lacks inherent visual semantic information. Second, the text needs to maintain legibility. Last, artistic text requires exquisitely designed interaction and layout to harmonize with surrounding text and background visual elements. Designing artistic text poses unique challenges that require specialized attention.

While there have been some works focusing on artistic text rendering, a comprehensive review of these techniques and analyses of their strategies to overcome the aforementioned challenges are still lacking. This paper aims to fill this gap by providing a comprehensive overview and analysis of the current

Figure 1: Artistic text generated by TET-GAN [115], ShapeMatching GAN [120], DS-Fusion [89] and Zou *et al.* [135]. Artistic text generation can be broadly classified into two categories: artistic text stylization and semantic typography.

state-of-the-art techniques in the field of artistic text rendering. We aim to provide researchers with a clear understanding of the development trajectory of this topic, its potential applications, available datasets, evaluation metrics, and future research directions.

The rest of the paper is organized as follows. We begin by defining and classifying the task of artistic text generation in Section 2. Next, Section 3 and Section 4 introduce the representative methods for artistic text stylization and semantic typography, respectively, providing detailed explanations of their main ideas, strengths, and weaknesses. Then, Section 5 discusses several application scenarios, and Section 6 introduces the existing artistic text datasets and evaluation metrics. Finally, we discuss future research directions in Section 7, and concluding remarks are given in Section 8.

2 Task Formulation

Artistic text generation is a conditional image generation problem. It typically involves a text input T and a style input S', intending to generate the corresponding artistic text T', preserving the shape of T while incorporating the style from S'.

In terms of the text input, T usually can be a text raster or vector image. With the recent development of cross-modality diffusion models [76], prompts can also be used as T to specify the desired text to be generated. For example, Stable Diffusion 3 demonstrates its powerful generative ability with an image generated using the prompt "Epic anime artwork of a wizard atop a mountain at night casting a cosmic spell into the dark sky that says 'Stable Diffusion 3' made out of colorful energy" as shown in Figure 2. Alternatively, in the field of NLP, natural language generation [50] could be exploited to produce the input text from more flexible conditions in various forms including text, image, table and knowledge base.

Figure 2: Artistic text generated by Stable Diffusion 3 with the prompt: Epic anime artwork of a wizard atop a mountain at night casting a cosmic spell into the dark sky that says "Stable Diffusion 3" made out of colorful energy. Image credits: Stable Diffusion 3 (https://stability.ai/news/stable-diffusion-3).

In terms of the style input, artistic text primarily encompasses two kinds of stylish effects: one is overlaid upon the text and the other is applied to the shape of the text itself. Therefore, artistic text generation can be divided into two major categories: artistic text stylization and semantic typography:

- **Artistic Text Stylization**. Artistic text stylization focuses on migrating visual effects (*i.e.*, text effects) from S' to T, including basic effects such as color, shadows, outlines, and gradients, as well as complex texture effects like flames and water ripples. Some examples are shown in Figure 1. Here, S' is usually a style image. Based on S', artistic text stylization can be further divided into two subcategories: text effect transfer, which directly imitates pre-designed text effects S' by artists, and arbitrary style transfer on text, which utilizes style elements from any image S' to design entirely new text effects and is more similar to standard image style transfer tasks, as illustrated in Figure 3.

(a) text effect transfer (b) arbitrary style transfer on text

Figure 3: Artistic text stylization can be divided into (a) text effect transfer and (b) arbitrary style transfer on text based on whether the style input S' is a well-designed text effect image or an arbitrary style image. Image credits: T-Effect [114] and UT-Effect [117].

- **Semantic Typography**. Semantic typography primarily deals with the shape of the text, aiming to deform the text into a target semantic content. For example, transforming the letter 'S' in the word "SNAKE" into the shape of a snake or transforming the entire word "umbrella" into the shape of an umbrella, as illustrated in Figure 1. Such shape transformations enable the text to visually match its intended meaning, enhancing its expressiveness and making the conveyed message more accessible even to those unfamiliar with the language.

It is worth noting that these two major approaches are not mutually exclusive. They can be combined to simultaneously modify the shape and apply visual effects to create fascinating artistic text artwork.

Meanwhile, according to the modality, the task can be divided into static artistic and dynamic text generation. Static artistic text generation primarily focuses on generating still images of artistic text. It involves applying various static text effects and style elements to the text, commonly used for creating visually appealing typographic designs, logos, posters, and other static visual compositions. In the context of the multimedia era, incorporating motion into artistic text has gained significant attention. Dynamic artistic text generation involves generating videos or GIF animations that showcase animated artistic text. In this case, there are two main aspects to consider: text effect animation and text shape animation.

- **Text Effect Animation**. Text effect animation focuses on studying the transfer of dynamic text effects onto the static text. In most cases, the text itself remains stationary. Sometimes, motion effects such as appearing, moving, scaling, or disappearing can also be considered as part of the text effects. In such a case, the output T' engages both text motions and animated visual effects.

- **Text Shape Animation**. Text shape animation primarily focuses on how to animate the semantic typography to resemble the motion of the intended semantic content naturally. For example, animating the

leg-kicking motion for the letter 'L' in the word "LEG". This involves animating the shape of the text to bring it to life and can visually convey more abstract concepts.

We summarize the taxonomy of the representative artistic text generation methods in Figure 4. Specifically, all methods can be roughly divided into artistic text stylization and semantic typography. In the task of artistic text stylization, two kinds of styles are considered: text effects and arbitrary style, corresponding to text effect transfer and arbitrary style transfer on text. Based on the modality, artistic text stylization can be further divided into static and dynamic artistic text stylization. Meanwhile, in the task of semantic typography, character-level and word-level design are studied. Based on the modality, semantic typography can be further divided into static semantic typography and kinetic typography. In the following sections, we will detail the main ideas of the representative methods in the order of their categories.

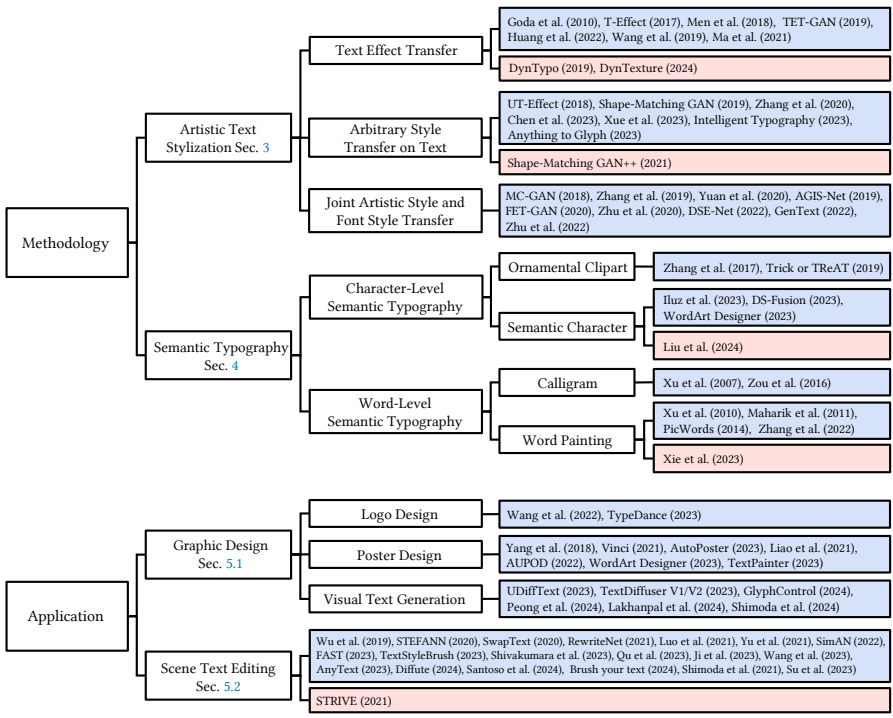


Figure 4: Taxonomy of the representative artistic text generation methods and applications. The blue background and red background indicate methods for static and dynamic artistic text generation, respectively.

Table 1: Summary of artistic text generation methods.

Method	Style type	Model type	Feature
Static Artistic Text Stylization			
Goda *et al.* [21]	calligraphy	stroke-based	ink texture synthesis along strokes
T-Effect [114]	text effects	patch-based	distribution-aware text effect prior
Men *et al.* [64]	text effects	patch-based	versatile interactive texture transfer
TET-GAN [115]	text effects	GAN-based	style-glyph disentanglement
Huang *et al.* [37]	text effects	GAN-based	simplified TET-GAN [115]
Wang *et al.* [98]	text effects	GAN-based	decorative element (decor) transfer
Ma *et al.* [61]	text effects	GAN-based	decor transfer on Chinese characters
UT-Effect [117]	arbitrary style	patch-based	structure transfer & texture transfer
Shape-Matching GAN [120]	arbitrary style	GAN-based	one-shot learning; style degree control
Zhang *et al.* [124]	arbitrary style	GAN-based	clean edges by erosion and dilation
Chen *et al.* [8]	arbitrary style	GAN-based	clean edges by erosion and dilation
Xue *et al.* [111]	arbitrary style	GAN-based	train a network to generate data
Intelligent Typography [63]	arbitrary style	GAN-based	coarse-to-fine complex style transfer
Anything to Glyph [96]	arbitrary style	diffusion-based	place objects according to the glyph
MC-GAN [4]	text effects & font	GAN-based	end-to-end stack network
Zhang *et al.* [129]	text effects & font	GAN-based	coarse-to-fine cascaded stack network
Yuan *et al.* [123]	text effects & font	GAN-based	text edge and skeleton as auxiliary input
AGIS-Net [19]	text effects & font	GAN-based	two parallel encoder-decoder branches
FET-GAN [51]	text effects & font	GAN-based	AdaIN-based text style modelling
Zhu *et al.* [132]	text effects & font	GAN-based	weighted style representation
DSE-Net [52]	text effects & font	GAN-based	effect-font-glyph disentanglement
GenText [36]	text effects & font	GAN-based	multi-task end-to-end training
Zhu *et al.* [133]	text effects & font	GAN-based	effect-font-glyph disentanglement
Dynamic Artistic Text Stylization			
DynTypo [65]	text effects	patch-based	global NNF search across frames
DynTexture [70]	text effects	patch & Transformer	long-distance dependency modeling
Shape-Matching GAN++ [119]	arbitrary style	GAN-based	spatial-temporal structural mappings
Static Semantic Typography			
Zhang *et al.* [125]	ornamental clipart	retrieval-based	joint semantic and shape matching
Trick or TReAT [90]	ornamental clipart	retrieval-based	unsupervised autoencoder matching
Iluz *et al.* [39]	semantic character	diffusion-based	vector glyph shape deformation
DS-Fusion [89]	semantic character	diffusion-based	raster semantic feature enhancement
WordArt Designer [26]	semantic character	LLM & diffusion	user-controllable artistic design
Xu *et al.* [107]	calligram	warp-based	shape adaptive text warping
Zou *et al.* [135]	calligram	warp-based	legibility enhanced calligram
Xu *et al.* [110]	word painting	structure-based	structural ASCII art generation
Maharik *et al.* [62]	word painting	vector field-based	adaptive text layout synthesis
PicWords [33]	word painting	warp-based	keyword semantic priority ranking
Zhang *et al.* [126]	word painting	vector field & SVM	visual saliency for aesthetic optimization
Kinetic Typography			
Liu *et al.* [57]	semantic character	diffusion-based	character deformation and animation
Xie *et al.* [104]	word cloud	frame-based	emotional word cloud animation

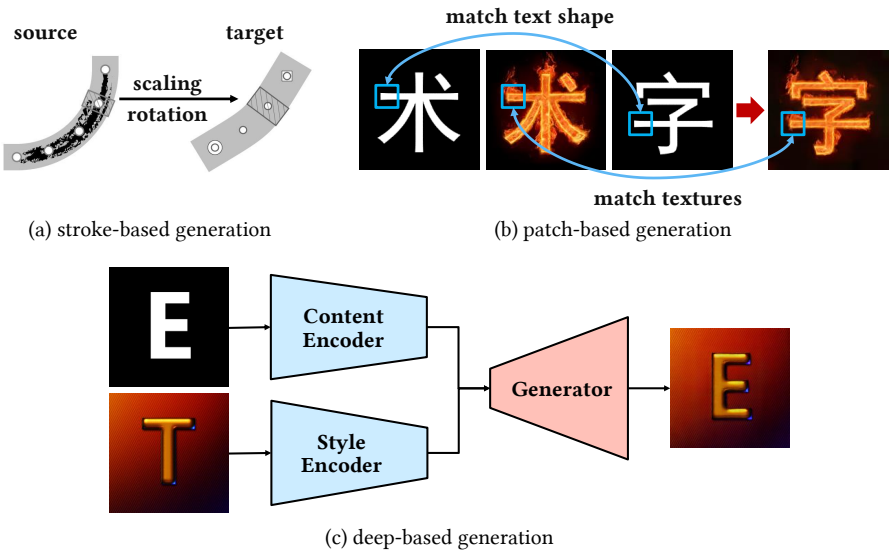
3 Artistic Text Stylization

Artistic text stylization involves the rendering of various visual effects, such as colors, shadows, outlines, gradients, textures, and embellishments, to enhance the aesthetics of text. First of all, text stylization pertains to image stylization, focusing on how to apply image stylization techniques to the specific content of text images. In general, the development of artistic text stylization has largely followed the trajectory of image stylization techniques. Chronologically, image stylization has moved from traditional texture synthesis to image translation leveraging deep neural networks and, most recently, to more sophisticated text-to-image techniques utilizing diffusion models. Concurrently, artistic text stylization approaches have incorporated specific designs into the above techniques to better transfer text effects and maintain the legibility of text. This section presents an overview of existing methods, highlighting their specific designs, strengths, and limitations.

3.1 Static Artistic Text Stylization

3.1.1 Text effect transfer

Stroke-based text effect transfer. Early image stylization methods primarily simulated how artists paint on the canvas: modeling brush strokes and synthesizing them onto a digital canvas [56, 29, 28]. Correspondingly, early text stylization methods focused on stylizing the strokes of characters, with the most representative research being calligraphy. In calligraphy, brush strokes vary in depth, pressure, and ink textures. Early studies [109] concentrated on synthesizing ink textures onto brush strokes. Directly applying texture synthesis technology [30] to characters often resulted in ink directions that did not align with the stroke directions. To address this, as illustrated in Figure 5(a), Goda *et al.* [21] propose to synthesize ink textures along the edges of characters by finding textures with lengths and directions that match the strokes, yielding more natural results.



(a) stroke-based generation

(b) patch-based generation

(c) deep-based generation

Figure 5: Different frameworks for artistic text stylization. Image credits: Goda *et al.* [21], T-Effect [114], TET-GAN [115].

Patch-based text effect transfer. Calligraphy represents only a small portion of text effects, characterized by relatively monochrome colors. To handle more diverse and colorful text effects, T-Effect [114] introduces the first style transfer method specifically for general text effects, such as shadows, outlines, gradients, and textures. In particular, T-Effect defines a supervised text effect transfer problem: $S : S' :: T : T'$ [30], where the source text effect

S' in addition to its corresponding plain text S are required. The algorithms learn the transformation between them and then apply it to the target text T to synthesize the result T'. Methodologically, T-Effect builds on the texture synthesis method of Wexler *et al.* [101] and its variants [12] using random search and propagation as in PatchMatch [5, 6]. As illustrated in Figure 5(b), the basic texture synthesis process is to match patches between S' and T' (to match textures), as well as S and T (to match text shape), and to update each patch in T' with its best-matched patch in S'. The process iteratively matches and updates patches until convergence. To extend this general texture synthesis to text effect, T-Effect analyzes real-world text effect images and summarizes a novel text effect prior: there is a high correlation between patch patterns (*i.e.*, color and scale) and their distances to text skeletons in high-quality text effects. This is because the artists commonly adjust the effects based on text shapes for readability. Based on this, T-Effect makes the following two modifications:

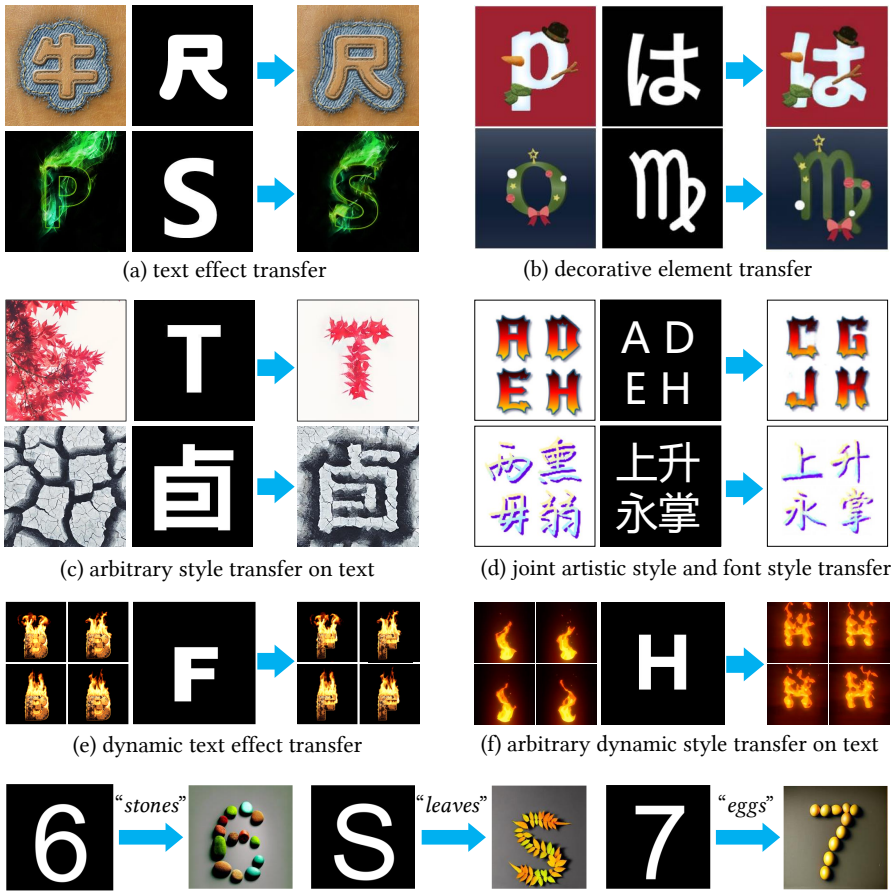
- Scale-adaptive matching: Encourages the patch to be matched at their optical scale based on their distance to the text skeletons. This could preserve both coarse structures and texture details.
- Distribution-aware matching: Encourages the text effects of T' to share similar distribution with S'. This could effectively realize spatial-aware style transfer.

Besides, the T-Effect further considers avoiding texture over-repetition for more natural synthesis. By incorporating priors specific to text effects into the patch match approach, T-Effect enables more plausible artistic text generation as in Figure 6(a).

Men *et al.* [64] extend T-Effect to more general interactive texture transfer applications, where S and T can be general semantic maps. Since semantic maps have large flat regions that provide few cues for valid matching, this method introduces novel structure guidance. Men *et al.* find that it is easier to build correspondences near the contours of the semantic maps. Based on the matched contour key points, the method propagates the structure guidance into inner flat regions to build rough correspondences based on which textures are synthesized. Thus, this method can better transfer inner textures than T-Effect.

However, the patch-based method requires iterative patch matching, resulting in a generation time of approximately one minute per image, which is not sufficiently efficient.

Deep-based text effect transfer. Entering the era of deep learning, researchers investigate the way of artistic text generation via data-driven learning. TET-GAN [115] is one of the first deep methods and enables real-time artistic text generation after training. TET-GAN builds a text effect



(a) text effect transfer
(b) decorative element transfer
(c) arbitrary style transfer on text
(d) joint artistic style and font style transfer
(e) dynamic text effect transfer
(f) arbitrary dynamic style transfer on text
(g) text-driven artistic text synthesis

Figure 6: Artistic text stylization results of representative methods. Image credits: (a) T-Effect [114]. (b) Wang *et al.* [98]. (c) ShapeMatching GAN [120]. (d) AGIS-Net [19]. (e) DynTypo [65]. (f) ShapeMatchingGAN++ [119]. (g) Anything to Glyph [96].

dataset with paired plain text images and the corresponding artistic text images. Then, the problem becomes learning a supervised image-to-image translation as in pix2pix [40]. However, directly applying pix2pix [40] can only learn one style at a time, which is less efficient in practice. To this end, TET-GAN introduces the idea of style-content disentanglement [46, 38, 79] into text effect transfer with separate content and style encoders, as shown in Figure 5(c). Specifically, TET-GAN builds a multi-task framework trained with three tasks:

- Glyph reconstruction: The network is trained to reconstruct plain text image T so that it can learn the glyph features.
- Artistic text destylization: The network is trained to infer the glyph features from the artistic text images S' so that it learns to disentangle the content representation.
- Artistic text stylization: The network is tasked to transfer the text effects of S' onto T, obtaining the output that approaches the ground truth T'. The network will learn to disentangle the style representation and combine it with the content representation for style transfer.

With this design, TET-GAN can learn hundreds of text effects in a single network. To further improve the practicality, it proposes a few-shot text effect fine-tuning strategy to efficiently extend the model to new text effects with only several and even one reference text effect image available.

Training on multiple tasks boosts the versatility of the model, which also increases the model complexity. Huang *et al.* [37] find that training a pix2pix network to map a channel-wise concatenated input of S, T and S' to the output T' is enough for multi-style transfer. This is useful when complicated functions like style extension and interpolation featured in TET-GAN are not required.

The aforementioned methods mainly treat text effects as a whole. However, some text effects have exquisite decor that needs special consideration. These decorative elements usually have very different styles from the base text effects. To address this problem, Wang *et al.* [98] propose to learn to separate, transfer and recombine the decors and the base text effects. The framework contains a network for decorative element segmentation, text effect transfer, and structure-based decor recomposition. During recomposition, the decorative elements are divided into two classes based on their importance. Insignificant elements are repeatable and randomly scattered on the text, while the significant elements are placed based on their spatial distributions in the original S'. The method can produce professional artistic typography on English letters and simple symbols as shown in Figure 6(b). Ma *et al.* [61] further extend this work to complex Chinese characters.

Although the aforementioned text effect transfer methods have achieved great success in synthesizing professionally artistic text images as in Figure 6(a)(b), they can only mimic the well-defined reference text effects. When it comes to more general arbitrary reference style images like the fire, water, and leaves (*e.g.*, Figure 6(c)(f)), as in common image style transfer tasks, these methods will fail. To handle these cases, researchers have paid attention to arbitrary style transfer on text.

3.1.2 Arbitrary style transfer on text

Text effect transfer is a well-defined problem with inputs S, S', and T, and the ground truth output T'. However, for arbitrary style transfer on text, we do not have the plain text version for S', and usually the ground truth output T' is not available, posing an unsupervised image-to-image translation problem for researchers. Furthermore, unlike text effects, there is a significant visual discrepancy between plain text and colorful style images. Therefore, this task is more challenging. The key is to find appropriate correspondences between the two distant domains.

Patch-based style transfer on text. UT-Effect [117] is one of the earliest methods to transfer arbitrary style onto text. To build valid correspondences between S' and T, it proposes extracting a binary mask S from S' based on texture removal [108], super-pixel extraction [1], and clustering. The region with higher saliency is set as the foreground corresponding to the text region in T, while the remaining region is the background. Then, the unsupervised problem becomes a supervised problem as proposed in T-Effect [114]. However, there is still an obvious structural discrepancy between S and T. For example, S might be maple leaves of different shapes, while T is the plain text with rigid edges. Directly synthesizing maple textures into T will result in unnatural maple boundaries. To solve this problem, UT-Effect proposes a two-stage style transfer framework:

- Structure transfer: UT-Effect transfers the structure styles of S onto T, obtaining $\hat{T}$ that shares similar boundaries with S while maintaining the glyph of T. To achieve this, UT-Effect proposes a legibility-preserving structure transfer method, which uses a patch-based shape synthesis technique [77] to adjust the shape of the stroke ends while preserving the shape of the stroke trunk for legibility.
- Texture transfer: For the translation problem $S : S' :: \hat{T} : \hat{T}'$, UT-Effect leverages the patch-based texture synthesis technique of T-Effect [114] and introduces a new saliency term to guide patch matching. The saliency term encourages pixels inside the text to find salient textures for synthesis and keeps the background less salient, which makes the artistic text stand out from the background.

GAN-based style transfer on text. Applying deep learning to arbitrary style transfer on text is challenging since there is generally no large-scale ground truth data for model training. To solve this problem, Shape-Matching GAN [120] proposes a one-shot bidirectional shape-matching framework to establish an effective glyph-style mapping at various deformation levels without paired ground truth. It includes two stages:

- Backward transfer: After extracting the structure map S from the style image S' as in UT-Effect [117], the first stage backward transfers the

shape style of the text to the structure map, obtaining its sketchy or simplified version $\bar{S}$, whose contour style is similar to the plain text.

- Forward transfer: The second stage learns the forward mapping from the sketchy structure map $\bar{S}$ to the original structure map S, and further to the original style image S'. The network learns to characterize the shape and texture features of the style image in the training phase and transfers these features to the target text in the testing phase.

To enable training on a single style image, Shape-Matching GAN randomly crops $\bar{S}$, S, and S' into sub-image pairs to obtain enough data. Another key contribution of Shape-Matching GAN is the style degree control mechanism. Specifically, there is a trade-off between legibility and artistry: The stylistic degree or shape deformations of a glyph need manipulation to resemble the style subject in S', while the glyph legibility needs to be maintained so that the stylized text is still recognizable. People's diverse preferences make it difficult to define an optimal style degree. Shape-Matching GAN introduces an extra parameter ℓ to control the style degree freely to allow users to select the most desired one. During backward transfer, ℓ is used to control the simplification degree of $\bar{S}$ to build multi-degree paired data, thus, during forward transfer, the model will learn multi-degree structure transfer conditioned on ℓ.

Based on Shape-Matching GAN [120], several improvements are proposed. Chen *et al.* [8] and Zhang *et al.* [124] use erosion and dilation to remove the unnecessary artifacts along the contour to generate cleaner structure transfer results. Zhang *et al.* [124] utilize multi-resolution style images borrowed from pyramid features [27] for better texture transfer. Xue *et al.* [111] directly train a dataset generation network to synthesize various paired data to overcome the problem of limited data.

Intelligent Typography [63] finds that it is hard for a $1\times$ network to learn a robust pixel-to-pixel level relationship from a single style image S' due to over-fitting. To overcome this issue in ShapeMatching GAN-based methods, Intelligent Typography develops a novel $2\times$ magnification network to smartly convert the problem of complex style transfer into texture expansion and super-resolution, which relieves the pressure of one-shot learning dramatically. To better transfer the style effects with relatively complex texture and structure, Intelligent Typography proposes a coarse-to-fine framework with two stages: prototype generation and detail refinement. Prototype generation generates a coarse-level stylized prototype from the given mask and tailored texture with the $2\times$ magnification network. Then, a structure network and a texture network are proposed to refine the details of the prototype. With the above design, Intelligent Typography can generate exquisite images with vivid artistic text details and clear backgrounds.

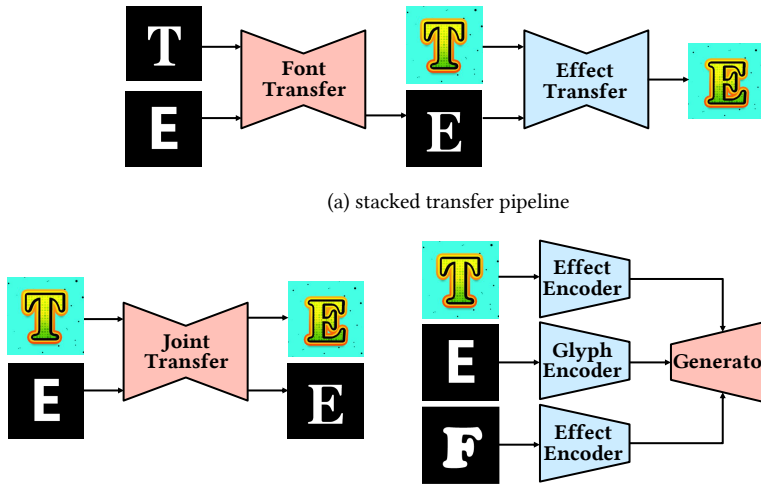
Diffusion-based style transfer on text. Recently, the increase in data scale and model representation capabilities has ultimately led to the emergence

of large diffusion models [76]. Diffusion models bring new opportunities for artistic text generation. Diffusion models exhibit unprecedented expressive power, providing diverse style support and interactive generation control with the help of large language models [72]. Anything to Glyph [96] is one of the recent diffusion-based, text-driven artistic text synthesis methods. It leverages the generative power of pre-trained Stable Diffusion [76] and segmentation models [58] to generate a paired dataset and train a diffusion model called Position Predictor to predict an object's position mask S in S'. Then, during testing, Denoising Score Matching [94] is applied to update the latent code of the Position Predictor so that its denoising result is similar to the target text shape T while maintaining consistency with the prompt. Given the updated latent code, which represents an initial structure transfer result (like $\hat{T}$ in UT-Effect [117]), pre-trained Stable Diffusion is used to synthesize textures onto it under the guidance of the prompt. Anything to Glyph is especially good at generating artistic text images composed of multiple instances of objects specified by the prompt such as stones, leaves, and eggs as shown in Figure 6(g).

3.1.3 Joint artistic style and font style transfer

Font is an important style of text complementary to the text effects. Different from text effects, large-scale text images with different fonts can be easily generated, which is suitable for deep learning. With the rapid development of deep generative models, font generation [3, 130, 106, 67, 97] has become a hot topic and has made great progress. In addition to large-scale supervised learning, researchers focus on investigating more challenging few-shot font generation [67, 97] and handwriting generation [25, 53]. Font generation can effectively design and produce new typefaces that can be used in various digital and print media. The generation process can automate the creation of font styles, weights, and variations, making it easier to produce large font families with consistent design characteristics. This section will briefly review representative approaches that combine font transfer and text effect transfer. High-quality joint transfer results are shown in Figure 6(d)

MC-GAN [4] is one of the first deep-based few-shot joint font and text effect transfer models. It contains a glyph network for font generation and an ornamentation network for text effect transfer. MC-GAN stacks these two networks and jointly trains them to realize an end-to-end solution, as illustrated in Figure 7(a). Specifically, the glyph network is pre-trained on a large-scale dataset so that it can predict the coarse glyph shapes of the missing English letters from a few stylized English letter examples. The full model with two networks is then fine-tuned on the stylized examples to learn to refine and stylize coarse glyph shapes into clean and well-designed letters.



(a) stacked transfer pipeline

(b) joint transfer pipeline

(c) disentangled transfer pipeline

Figure 7: Different frameworks for joint artistic style and font style transfer. Image credits: TET-GAN [115].

Zhang *et al.* [129] further divide the glyph network of MC-GAN [4] into a coarse-level image-to-image translation network and a fine-level stack network. In terms of text effect transfer, it utilizes the widely used Neural Style Transfer [20]. On the other hand, Yuan *et al.* [123] find that using edge and skeleton information of T as auxiliary input of the glyph network could better infer the shape of the font.

AGIS-Net [19] proposes to disentangle the content and style representations with two encoders and two decoders. Two encoders learn to extract the content feature and style feature, respectively. Then two collaborative decoders generate the glyph shape image and final artistic text image simultaneously based on the extract features. The disentanglement ensures a few-shot multi-content and multi-style generation.

Inspired by the AdaIN-based style representation [35], FET-GAN [51] views font style and text effect style as a whole and models them with channel-wise means and standard deviations as in Huang *et al.* [35]. The content feature extracted from the source image is stylized through AdaIN operation and further decoded to obtain the final stylized artistic text image. The pipeline is shown in Figure 7(b).

Zhu *et al.* [132] also propose a few-shot end-to-end framework that extracts and combines content and style representations for joint font and text effect transfer. The key idea is that text content and style are not fully independent. Therefore, it calculates the similarity between the reference glyph and the

target glyph to assign weights for the style features of each referenced glyph. Then, the weighted style representation and the content representation are fused to generate the final artistic text image.

DSE-Net [52] argues that treating font style and text effect style as a whole would limit the transfer of complex styles. It presents a disentangled style encoding network as illustrated in Figure 7(c), with three different networks to extract the font feature, text effect feature, and glyph content feature, respectively. Finally, a cross-layer fusion mechanism is proposed to fuse the features adaptively to generate the final output.

GenText [36] extends TET-GAN [115] with an extra task of font transfer. It treats the plain text as an artistic text image with special text effects, thus unifying the destylization and stylization tasks within a single network. Then, GenText uses an encoder network to extract the content code and the style code from artistic text images, a font transfer network to fuse the content code of the plain text and the style code of the plain text with reference font for font transfer, and a text effect transfer network to fuse the content code of the plain text and the style code of the artistic text for stylization and destylization. Then, artistic text can be generated by sequentially performing font transfer and stylization.

Zhu *et al.* [133] propose an artistic text style transfer model based on multi-factor disentanglement and mixture. The model contains three encoders to extract the text effect, font, and glyph representations, and further employs adversarial training and one-factor swap training strategies to disentangle the three representations. The disentanglement enables several tasks of font transfer, text effect transfer, joint transfer, and style removal.

3.2 Dynamic Artistic Text Stylization

Compared to static artistic text, dynamic artistic text is more attractive and is widely used in a variety of media such as films, advertisements, and video clips. While static artistic text stylization [114, 117, 115, 120] and general video image style transfer [80, 24, 35, 7, 99] have been extensively studied, a few approaches [65, 70, 119] have studied dynamic artistic text stylization, which is reviewed in this section.

Different from general video style transfer that migrates static style from a style image onto a content video, dynamic artistic text stylization aims to transfer dynamic style from a style video $\mathbf{S}' = \{S'_1, S'_2, ..., S'_N\}$ onto a text image T as illustrated in Figure 6(e)(f), where N is the total frame number. Therefore, it is not straightforward to apply the optical flow guidance [80, 24, 35, 7] widely used in video style transfer to dynamic artistic text generation. Instead, dynamic artistic text stylization approaches mainly focus on spatial-temporal style representation modeling and transfer.

3.2.1 Dynamic text effect transfer

DynTypo [65] extends the NNF search of PatchMatch [5] to the spatial-temporal domain. Instead of searching the Nearest-neighbor Field (NNF) for text effect synthesis in a frame-by-frame manner, the main idea of DynTypo is to simultaneously optimize the text effect coherence across all frames to find a common NNF for all temporal frames. Specifically, DynTypo stacks patches at the same position but across an entire video into a patch cube and matches patches at a cube level. After matching, the entire cubes are directly used to synthesize the output video. DynTypo first detects keyframes based on the intensity of text effect dynamics and then limits the procedure of cube matching within the keyframes to maintain both spatial and temporal consistencies. DynTypo further combines PatchMatch with Simulated Annealing, to add more priority to the patches near the text contours. With the above designs, DynTypo achieves impressive results (*e.g.*, Figure 6(e)) in dynamic text effect transfer.

However, it is hard for a single global NNF to transfer complex dynamic text effects such as moving samples that shift across source videos. To better model the inter-frame correlation and transfer complex dynamic text effects, DynTexture [70] proposes to combine PatchMatch [5] with the advanced Transformers [93]. Specifically, DynTexture decomposes the dynamic text effect transfer task into two stages.

- First frame generation: DynTexture adopts patch-based text effect transfer [114] with distance map guidance [65] to generate the first frame.
- Full video generation: The synthesized first frame is decomposed into structure-agnostic patches, which are then encoded to tokens. Then, Transformers [93] equipped with VQ-VAE [92] are exploited to predict the discretized token sequences, leveraging its high capability of capturing the long-distance dependencies between frames. All predicted patches are assembled into each frame by a Gaussian weighted average merging strategy to obtain the final full video result.

3.2.2 Arbitrary dynamic style transfer on text

In the scope of arbitrary style transfer on text, Shape-Matching GAN++ [119] extends Shape-Matching GAN [120] to video domains. Shape-Matching GAN learns the forward mapping from the sketchy structure map $\bar{S}$ to the original structure map S at a patch level. In Shape-Matching GAN++, to learn spatial-temporal shape mappings, patches across consecutive frames are learned together, just as the patch cude in DynTypo [65], so that the model could learn inter-frame motion patterns. To generate long-term motions, Shape-Matching GAN++ divides the full video into multiple K-frame video clips and focuses on

the short-term motion patterns of these video clips. It defines the short-term motion pattern as the shape dynamics between the frames within a video clip. During training, the model learns the shape mappings between the first $K-1$ frames and the last frame of a video clip, which is actually a frame prediction task. During testing, the model repeatedly predicts the next frame based on previous frames and propagates the short-term motion patterns to achieve long-term motion patterns. Figure 6(f) presents an example of the dynamic style transfer result by ShapeMatching GAN++.

4 Semantic Typography

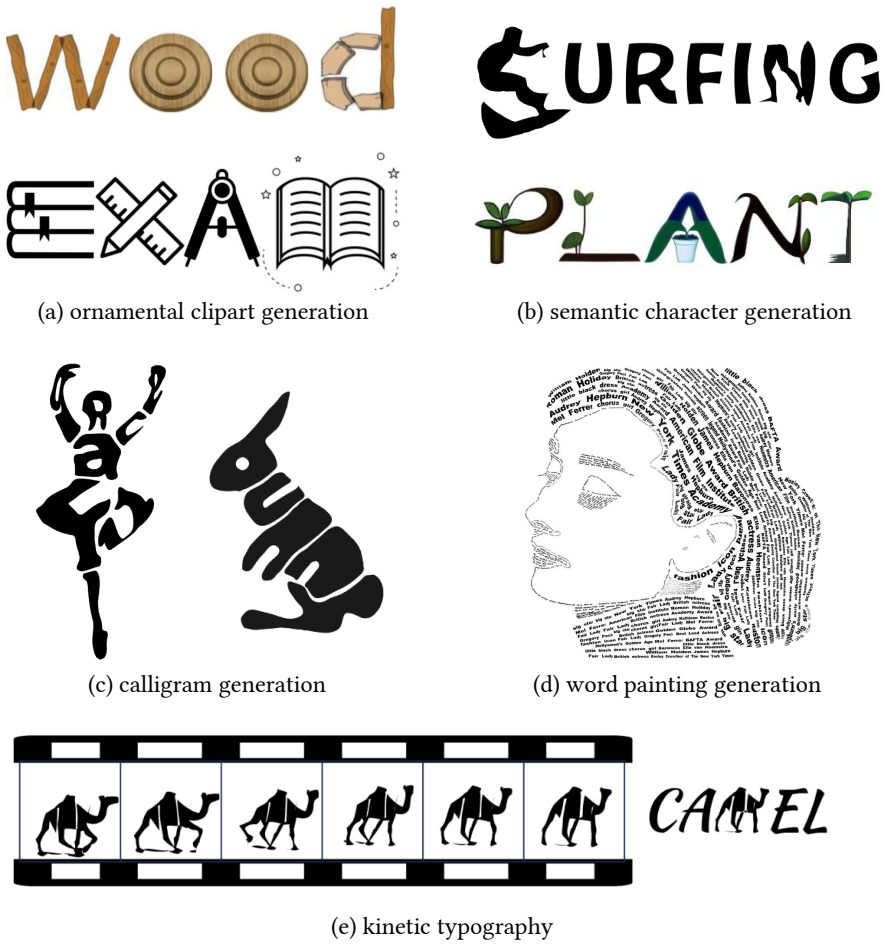
Semantic Typography focuses on transforming the text shape to visually represent specific objects, themes, or concepts. This section investigates different techniques, including shape morphing, deformation, and adaptive typography, to realize this meaningful and visually appealing art form.

4.1 Static Semantic Typography

4.1.1 Character-level semantic typography

Ornamental Clipart Generation. Ornamental clipart generation aims to identify suitable images that correspond to the semantics of words and meticulously assemble them under the direction of glyph strokes as shown in Figure 8(a), which is considerably time-consuming when carried out manually. Zhang *et al.* [125] present an automatic framework for creating such ornamental typefaces. Specifically, the framework features an interactive interface for the glyph stroke segment with scribbles from the user, thus fulfilling their personalized requirements. In terms of image selection, a semantic-shape similarity metric is established to concurrently account for both word semantics and stroke form. An optional structural optimization step based on gradient descent is implemented to yield results with enhanced glyph structure and aesthetic appeal.

Trick or TReAT [90] is another retrieval-based method for ornamental clipart generation. It leverages clipart to resemble the glyphs of characters to express the semantic features of words. To better retrieve the similarity between letters and clipart, it trains an auto-encoder based on AlexNet [44] in an unsupervised manner to automatically learn a hidden feature space. The corresponding clipart will be retrieved based on the distance in the hidden space. Trick or TReAT generates results that have good legibility, semantics, and creativity. However, as with all retrieval-based methods, the diversity of the output is inherently dependent on the input data, which may introduce some limitations in the richness of the generated content.



(a) ornamental clipart generation

(b) semantic character generation

(c) calligram generation

(d) word painting generation

(e) kinetic typography

Figure 8: Semantic typography results of representative methods. Image credits: (a) Zhang *et al.* [125], Trick or TReAT [90]. (b) Iluz *et al.* [39], DS-Fusion [89]. (c) Xu *et al.* [107], Zou *et al.* [135]. (d) Zhang *et al.* [126]. (e) Liu *et al.* [57].

Semantic Character Generation. With the development of generative models in recent years, diffusion models have demonstrated exceptional performance in image synthesis tasks [75, 31, 76, 81, 74]. The robust cross-modal understanding and rich semantic priors encapsulated make diffusion models highly adaptable for semantic typography. As shown in Figure 8(b), Iluz *et al.* [39] pioneer the integration of a pre-trained Stable Diffusion model into the semantic character generation process. Focusing on the geometric transformation of glyphs, they employed the SDS loss [69] and DiffVG to tailor diffusion

models for the artistic vectorized glyph shape generation. Additionally, they incorporated the as conformal as possible (ACAP) loss [32] to regulate the extent of character deformation and established a tone preservation loss to maintain the structural integrity of the original glyph. This approach not only enhances the adaptability of diffusion models but also ensures that the generated characters retain their distinctive aesthetic and semantic qualities.

In contrast to the approach of Iluz *et al.* [39], DS-Fusion [89] places a greater emphasis on texture and color features by directly processing glyph images in raster form. Guided by style prompts and glyph images, DS-Fusion integrates the glyph shape with style images derived from a pre-trained Stable Diffusion model [76], which is conditioned with style prompts. A CNN-based discriminator is employed to provide implicit supervision of the generation process by examining the feature maps of both the glyph shape and the generated image in latent space. It is noteworthy that DS-Fusion, when supplied with an entire word as a glyph reference, is also adept at producing word-level stylized results that exhibit harmonious character combinations and a vivid artistic expression.

WordArt Designer [26] presents a user-controllable artistic semantic character design system with a large language model (LLM) engine [2]. It contains three modules SemTypo, StyTypo, and TextTypo. SemTypo employs character parameterization and rasterization techniques akin to those used by Iluz *et al.* [39], to manipulate semantic features and deform selected parts of character strokes. StyTypo refines the smoothness and stylization details by leveraging the depth2image approach of latent diffusion models [76]. In TextTypo, a ControlNet [128] is conditioned with Canny edges, depth maps, scribbles, and the original text image, enabling it to render textures that align with semantic styles and glyphs. The LLM engine generates structured text prompts based on user descriptions, feeding into the aforementioned models and thereby amplifying the creative diversity.

4.1.2 Word-level semantic typography

Calligram Generation. A calligram is a creative arrangement of words or letters that forms a visual image, conveying both meaning and aesthetics, as shown in Figure 8(c). Xu *et al.* [107] first introduce a warp-based method to integrate letters into the subject region of an image, thereby crafting calligrams that possess semantic features alongside logical glyph stroke deformation. This method employs an interactive approach to divide the container into subregions, which are subsequently filled by warping letters. While this approach can produce calligrams with coherent letter shape arrangements, it sometimes struggles with legibility.

To address this issue, Zou *et al.* [135] conduct a crowd-sourced study aimed at refining the guidance for glyph deformation to enhance letter legibility. They introduced a fully automatic method for generating letter layouts, aligning letters based on correspondence, and applying deformation.

Word Painting Generation. Word painting represents a form of composite artwork, characterized by the adaptive fusion of visual structure and texture derived from a source image with semantic features extracted from a textual source, as shown in Figure 8(d). Xu *et al.* [110] formulate this task as approximating the primary structure of a reference image using ASCII characters. This method surpasses traditional tone-based techniques by capturing structural and semantic nuances through an innovative alignment-insensitive shape similarity metric. Maharik *et al.* [62] introduce a technique for crafting micrographics, a distinctive variant of word painting that comprises minuscule letters. Departing from the conventional single horizontal text layout of ASCII art, this method emphasizes the design of low curvature and smooth vector fields devoid of singularities, facilitating the synthesis of word layouts that conform to regional shapes while maintaining high text readability. Additionally, a warping procedure for text height and width is proposed, enabling the adjustment of text shape to seamlessly integrate with the image contours.

PicWords [33] is an automatic word painting generation framework that employs non-photorealistic rendering (NPR) techniques. It processes the input image by segmenting the binary silhouette into distinct patches, each designed to encapsulate a keyword. By ranking the patches and keywords, this approach establishes a keyword-patch correspondence that serves to accentuate significant keywords, ensuring that key semantic information is effectively conveyed. Zhang *et al.* [126] introduce a method that utilizes a smooth vector field for patch segmentation, coupled with a Support Vector Machine (SVM)-based visual attention model to optimize the aesthetic arrangement of text. This visual attention model is trained to generate a saliency map for a given image, strategically positioning keywords that are closely related to the theme in areas that naturally draw the viewer's attention. Compared with prior work [62], this approach is capable of producing results that are imbued with more profound semantic information, enhancing both the visual appeal and the narrative depth of the generated word paintings.

4.2 Kinetic Typography

Kinetic typography is a dynamic and expressive art form that combines motion graphics with typography to convey emotions, narratives, or messages in a visually engaging way. It involves animating text so that it moves, transforms, or interacts with other elements in a sequence or scene. Early works focus on exploring the interaction between dynamic forms and content to enrich emotional expression [15, 48], with various kinetic typography system designs

for animating text [47, 16, 66]. Wakey-Wakey [105] presents an automatic framework for aligning dynamic text motions with GIF animation. However, these methods lack effective measures to combine semantic features with textual dynamics.

Liu *et al.* [57] introduce a novel approach to generate text animation with semantic features. In contrast to the work of Iluz *et al.* [39], they have developed an end-to-end model designed to mitigate conflicts with prior knowledge. As shown in Figure 8(e), this model leverages neural displacement fields and vector representations to deform letters, thereby conveying semantic meanings and rendering them in dynamic movements that respond to user prompts. Xie *et al.* [104] propose a method to animate compact word clouds, expanding the scope from single-word animations to multiple words that express semantic emotions.

5 Applications

5.1 Graphic Design

WordArt and LOGO design is a task involving the transformation of text input into semantically rich typography, incorporating various elements and layouts, as shown in Figure 9(a). Numerous studies have successfully addressed the generation of diverse WordArt and LOGO designs, utilizing a wide range of elements and layouts [26, 100, 103].

Similarly, poster design entails the arrangement and styling of provided text, subsequently generating the designed text on an input image, shown in Figure 9(b). Several works have achieved advancements in this area [116, 23, 55, 54, 34, 26, 18].

Visual text generation focuses on producing accurate and legible text within image generation, addressing the common limitation of image generation models, which often struggle to create readable text, shown in Figure 9(c). Various studies have achieved notable success in generating images with clear and coherent text [131, 11, 10, 121, 68, 45, 85].

5.2 Scene Text Editing

Scene text editing has emerged as a notable research area in recent years, focusing on the transformation of text within a source image to align with a target reference text while preserving the original style and background, shown in Figure 9(d). Numerous studies have proposed methodologies to generate high-quality text images despite varying backgrounds and fonts [102, 78, 113, 49, 60, 122, 59, 13, 43, 86, 71].

Diffusion models have demonstrated significant potential in editing images of arbitrary topics, including scene text. To fully exploit this potential, several

(a) WordArt and LOGO design (b) poster design

(c) visual text generation

(d) scene text editing

Figure 9: The examples of applications. (a) WordArt and LOGO design by WordArt Designer [26], Wang *et al.* [100]. (b) Poster design by Vinci [23], AUPOD [34]. (c) Visual text generation by GlyphControl [121], TextDiffuser-2 [10]. (d) Scene text editing by Qu *et al.* [71], SwapText [113], FAST [13].

works have applied diffusion models to the task of scene text editing [41, 95, 91, 9, 83, 127].

The subfield of scene-style text editing, addressed in some research [84, 87], entails modifying specific attributes of text within an image while either preserving or altering its style. These attributes include rotation, font, color, and content, allowing for versatile text manipulations across various images.

Moreover, scene text editing can be extended to video applications [88]. Enhancing scene text editing in videos necessitates preserving geometric integrity, appearance consistency, and temporal coherence.

6 Dataset and Evaluation

6.1 Datasets

As summarized in Table 2, there are multiple available datasets for artistic text rendering and design. As shown in Figure 10, these datasets mainly differ in several major aspects: 1) whether font styles are taken into consideration, 2) whether the reference text style contains special elements, 3) the kinds of character types that are involved, and 4) the usage scenarios in the real world.

Table 2: Summary of the benchmark datasets for artistic text rendering and design

Dataset	Type	Images	Feature
MC-GAN-Gray [4]	joint font & text effects	260,000	gray-scale English letter.
MC-GAN-Color [4]		520,000	colorful English letter.
AGIS-Net-C [19]	joint font & text effects	1,571,940	synthetic artistic Chinese characters.
AGIS-Net-P [19]		256,410	professional-designed artistic Chinese characters.
TET-GAN [115]	text effects	53,568	64 text effects rendered on 775 Chinese characters, 52 English letters and 10 Arabic numerals.
TextEffects-Decor [98]	text effects	59,280	64 text effects with decorative elements rendered on 52 English letters of 19 fonts.
TE141K-E [118]	text effects	66,196	64 text effects on 52 English letters of 19 fonts.
TE141K-C [118]		54,405	65 text effects rendered on 775 Chinese characters, 52 English letters and 10 Arabic numerals.
TE141K-S [118]		20,480	20 text effects rendered on 56 special symbols, and 968 letters in Japanese, Russian, *etc.*
SSAF-CN [52]	text effects	97,200	100 text effects rendered on 972 Chinese characters.
SSAF-EN [52]		2,600	100 text effects rendered on 26 English letters.
Imgur5K [43]	handwriting	135,375	135,375 handwritten English words from 5,305 images
TextLogo3K [100]	text logo	3,470	text logo images extracted from poster/covers of movies, TV series and comics.
MARIO-10M [10]	visual text	10,061,720	9,194,613, 343,423 and 523,684 text images from natural images, posters, and book covers, respectively.
LAION-Glyph [121]	visual text	∼10,000,000	images with rich visual text content
AnyWord-3M [91]	visual text	3,034,486	text images from several datasets 1.6 million in Chinese, 1.39 million in English, and 10k images in other languages.

For the first aspect, the first four datasets, namely MC-GAN-Gray [4], MC-GAN-Color [4], AGIS-Net-C [19], and AGIS-Net-P [19], consider fonts along with text effects. For the second aspect, only TextEffects-Decor [98] collects effects with decorative elements. For the third aspect, TET-GAN [115], TE141K-C [118], TE141K-S [118], and AnyWord-3M [91] contain characters of multiple languages, while others only collect either English letters or Chinese characters. For the last aspect, Imgur5K [43], TextLogo3K [100], MARIO-

Figure 10: Samples of the datasets for artistic text rendering and design.

10M [10], LAION-Glyph [121], and AnyWord-3M presenting text images from various life circumstances, spanning aspects of advertising billboards, movie posters, book covers, handwritten words, internet memes, *etc.*

In the following, we briefly introduce the details of the representative datasets for artistic text rendering and design.

(1) MC-GAN-Gray [4] is a joint font and text effects dataset that includes 10K gray-scale Latin fonts each with 26 capital English letters, a total of 260,000 images. These images were processed by bounding boxes, and each glyph was standardized and placed at the center of a 64×64 image. Excluding font ornamentation, these fonts provide extensive information about inter-letter correlations in font styles, encoding only the glyph outlines. Based on the former, Azadi *et al.* apply random color gradients and outlining onto the gray-scale glyphs, resulting in a dataset of 20K color fonts and a total of 520,000 images, namely MC-GAN-Color [4].

(2) AGIS-Net-C [19] dataset is a joint font and text effects dataset that includes 246 normal Chinese font styles and 10 different kinds of gradient colors and stripe textures, with 639 representative Chinese characters, a total of 1,571,940 images. For few-shot tasks, Gao *et al.* also introduce the AGIS-Net-P [19] dataset, which consists of 35 selected professional-designed fonts with 7,326 Chinese characters for each style, a total of 256,410 images.

(3) TET-GAN [115] dataset is a text effects dataset that includes 64 text effects each with 775 Chinese characters, 52 English letters, and 10 Arabic numerals, a total of 53,568 images. Each text effects image, sized at 320 × 320, is accompanied by its corresponding plain text image.

(4) TextEffects-Decor [98] is a text effects dataset that includes 60 text effects on 52 English letters of 19 fonts, a total of 59,280 images. Each text effect has a size of 320 × 320 and is provided with its corresponding raw text, and decorative elements were collected from www.shareicon.net.

(5) TE141K [118] is a text effects dataset that includes 152 text effects rendered on glyphs (English letters, Chinese characters, and Arabic numerals). Each text effect has a resolution of 320 × 320 and is provided with its corresponding glyph image. Based on glyph types, TE141K was divided into three subsets: TE141K-E contains 67 styles and 988 glyphs, a total of 66,196 image pairs of English alphabets; TE141K-C contains 65 styles and 837 glyphs, a total of 54,405 image pairs of Chinese characters; TE141K-S contains 20 styles and 1,024 glyphs, a total of 20,480 image pairs of special symbols and letters from common languages other than Chinese and English.

(6) SSAF [52] is a text effects dataset that includes 200 text effects rendered on glyphs (Chinese characters and English letters). Each text effect has a resolution of 320 × 320 and is provided with its corresponding glyph image. Based on glyph types, SSAF was divided into two subsets: SSAF-CN contains 100 Chinese artistic fonts, each with 972 Chinese characters; SSAF-EN contains 100 English artistic fonts, each with 26 uppercase English letters.

(7) Imgur5K [43] is a handwriting image dataset that includes 135,375 handwritten English words from 5K images originally hosted publicly on Imgur.com. Each image was assigned to no more than five annotators, and spurious data was eliminated by the utility of the annotation averages of word bounding boxes and the highest agreement on labeled content strings.

(8) TextLogo3K [100] is a text logo dataset that includes 3,470 text logo images of posters and covers of movies, TV series, and comics which were selected from Tencent Video, one of the leading online video platforms in China. Each character has been annotated by bounding box, pixel-level mask, category, and the angle of rotation and affine transformation (if existing).

(9) MARIO-10M [10] is a visual text dataset that includes 10,061,720 image-text pairs of natural images, posters, and book covers, each with comprehensive OCR annotations. Based on the difference in data sources,

MARIO-10K was divided into three subsets: MARIO-LAION contains 9,194,613 high-quality text images with corresponding captions, covering a broad spectrum of advertisements, notes, posters, covers, memes, logos, *etc*; MARIO-TMDB contains 343,423 English posters from The Movie Database (TMDB), a community-built dataset for movies and TV shows with high-quality posters; MARIO-OpenLibrary contains 523,684 original-size covers from Open Library, an open and editable library catalog that generates a web page for every published book.

(10) LAION-Glyph [121] is a visual text dataset that includes about 10M images with rich visual text content which were selected by using a modern OCR system. The amount of characters in the images primarily ranges from 10 to 50, with most samples containing fewer than 150 characters. Based on the considerations for the convenience of training and evaluation, the LAION-Glyph dataset was divided into three scales: LAION-Glyph-100K, LAION-Glyph-1M, and LAION-Glyph-10M.

(11) AnyWord-3M [91] is a visual text dataset that includes 3,034,486 multi-language scene text images, covering street views, book covers, advertisements, posters, movie frames, *etc*. AnyWord-3M contains approximately 1.6M images in Chinese, 1.39M images in English, and 10k images in other languages, spanning Japanese, Korean, Arabic, Bengali, and Hindi.

6.2 Performance Evaluation

For text style transfer tasks with paired output and ground truth, instance-level similarities such as L1 loss, MSE loss, peak signal-to-noise ratio (PSNR), structural similarity index (SSIM), and perceptual loss [42] can be used to calculate differences between the output image and the ground truth image. Meanwhile, on the tasks where the ground truth artistic text images are not available for each input raw text image, distribution similarity could be explored. Inception score (IS) [73] and Fréchet inception distance (FID) [82] are introduced to measure the distribution distance between the generated images and the ground truth dataset. However, the aforementioned evaluation metrics are mostly designed for natural images, thus limiting the evaluation ability of artistic text.

To tailor the evaluation metrics to the realm of artistic text, content- and style-related metrics are introduced to this task. Style loss [42] and CLIP score [72] are widely used in semantic character generation and visual text generation for further assessing the expression of semantic information, which is effective for evaluating tasks that emphasize output artistry. For tasks like arbitrary style transfer on text and character-level semantic typography where the glyph is adjusted, the legibility of the text is crucial. To evaluate the text legibility and correctness, OCR accuracy, Sentence Accuracy, and Text

Detection Accuracy are mainly used in scene text editing tasks [91, 127, 113, 71, 102]. Yan *et al.* [112] provide a novel assessment system specially focused on artistic text stylization. A multi-task network is proposed to extract features of artistic text images and is trained on selected data from TE141K [118] with user labels, thus imitating visual evaluation from humans. This model stands as a robust instrument to assist the quality assessment process.

Qualitative evaluation methods such as user study are effective ways to analyze the quality and aesthetic appeal of the generated result by human aesthetic judgment. However, the result could be influenced by the preference of individual participants, which gives it a certain degree of subjectivity and makes it difficult to reproduce.

7 Future Challenges

Although the rapid advancement of artificial intelligence and deep learning has made great progress in artistic text generation and design, and the recent methods can generate satisfactory results, there are still several challenges and open issues. This section discusses some key challenges in artistic text generation.

One of the challenges in the automatic generation of artistic text is that current methods stylize text best based on concrete visual concepts. It is still difficult to use abstract concepts to influence the stylization process. This limitation hinders the ability to fully leverage the creative potential of abstract concepts and ideas in the artistic rendering of text. One possible solution is to leverage large language models (LLMs) to rephrase the abstract concepts into more descriptive ones.

Another challenge is that current methods are mostly based on diffusion models. However, the sampling process of diffusion models is well-known to be inherently slow, leading to a slow generation of artistic text. A possible solution is to explore faster approximation methods or efficient sampling techniques. Additionally, model distillation could further reduce generation time.

While significant progress has been made in static artistic text generation with various emerging methods, dynamic artistic text generation remains under-explored. This is partly due to the lack of sufficient video data and the increased complexity of generating video compared to images. One great demand is to develop large, high-quality datasets specifically for dynamic artistic text. Additionally, employing advanced video generation techniques, such as temporal consistency models and leveraging transfer learning from static to dynamic scenarios, could help address the challenges in this area.

Another challenge in artistic text generation is achieving fine-grained control. While diffusion-based methods can generate diverse artistic text, text guidance offers coarse-grained control over the output. It is difficult to achieve

fine-grained control, such as changing specific regions or styles of the text, adjusting individual character shapes, adapting the text to different font sizes, or flexibly modifying the artistic degree to balance artistry and legibility. A potential solution is to integrate attention mechanisms and region-specific conditioning into the models. Additionally, developing interactive tools that allow users to manually adjust and refine specific aspects of the generated artistic text can also enhance fine-grained control and customization.

8 Conclusion

This paper provides a comprehensive survey of artistic text rendering and design. We first explored two main categories: artistic text stylization and semantic typography, along with the incorporation of motion for dynamic artistic text generation. Second, several applications in the context of artistic text generation are detailed. Third, we studied the datasets and evaluation of the artistic text. Finally, several challenges and future research directions were discussed. The need for creative, efficient, versatile, and controllable generative models was emphasized. We hope this paper can serve as a foundation for further advancements and inspire the exploration of new avenues in this exciting field.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Grant 62332010, in part by the CCF-Tencent Open Research Fund, and in part by the Key Laboratory of Science, Technology and Standard in Press Industry (Key Laboratory of Intelligent Press Media Technology).

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC Superpixels Compared to State-of-the-Art Superpixel Methods", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012.

[2] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, *et al.*, "Gpt-4 technical report", *arXiv preprint arXiv:2303.08774*, 2023.

[3] G. Atarsaikhan, B. K. Iwana, A. Narusawa, K. Yanai, and S. Uchida, "Neural font style transfer", in *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*, Vol. 5, IEEE, 2017, 51–6.

[4] S. Azadi, M. Fisher, V. G. Kim, Z. Wang, E. Shechtman, and T. Darrell, "Multi-content gan for few-shot font style transfer", in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2018.

[5] C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman, "PatchMatch: a Randomized Correspondence Algorithm for Structural Image Editing", *ACM Transactions on Graphics*, 28(3), August 2009, 341–52.

[6] C. Barnes, E. Shechtman, D. B. Goldman, and A. Finkelstein, "The Generalized Patchmatch Correspondence Algorithm", in *Proc. European Conf. Computer Vision*, 2010, 29–43.

[7] D. Chen, J. Liao, L. Yuan, N. Yu, and G. Hua, "Coherent Online Video Style Transfer", in *Proc. Int'l Conf. Computer Vision*, 2017, 1105–14.

[8] F. Chen, Y. Wang, S. Xu, F. Wang, F. Sun, and X. Jia, "Style transfer network for complex multi-stroke text", *Multimedia Systems*, 29(3), 2023, 1291–300.

[9] H. Chen, Z. Xu, Z. Gu, Y. Li, C. Meng, H. Zhu, W. Wang, *et al.*, "Diffute: Universal text editing diffusion model", *Advances in Neural Information Processing Systems*, 36, 2024.

[10] J. Chen, Y. Huang, T. Lv, L. Cui, Q. Chen, and F. Wei, "TextDiffuser-2: Unleashing the Power of Language Models for Text Rendering", *arXiv preprint arXiv:2311.16465*, 2023.

[11] J. Chen, Y. Huang, T. Lv, L. Cui, Q. Chen, and F. Wei, "Textdiffuser: Diffusion models as text painters", *Advances in Neural Information Processing Systems*, 36, 2024.

[12] S. Darabi, E. Shechtman, C. Barnes, D. B. Goldman, and P. Sen, "Image Melding: combining Inconsistent Images Using Patch-based Synthesis", *ACM Transactions on Graphics*, 31(4), July 2012, 82:1–82:10.

[13] A. Das, P. Roy, S. Bhattacharya, S. Ghosh, U. Pal, and M. Blumenstein, "FAST: Font-Agnostic Scene Text Editing", *arXiv preprint arXiv:2308.02905*, 2023.

[14] A. A. Efros and W. T. Freeman, "Image Quilting for Texture Synthesis and Transfer", in *Proc. 28th Annual Conf. Computer Graphics and Interactive Techniques*, 2001.

[15] S. Ford, J. Forlizzi, and S. Ishizaki, "Kinetic typography: issues in time-based presentation of text", *CHI '97 Extended Abstracts on Human Factors in Computing Systems*, 1997.

[16] J. Forlizzi, J. Lee, and S. Hudson, "The kinedit system: affective messages using dynamic texts", in *Proceedings of the SIGCHI conference on Human factors in computing systems*, 2003, 377–84.

[17] O. Frigo, N. Sabater, J. Delon, and P. Hellier, "Split and match: Example-based adaptive patch sampling for unsupervised style transfer", in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2016, 553–61.

[18] Y. Gao, J. Lin, M. Zhou, C. Liu, H. Xie, T. Ge, and Y. Jiang, "TextPainter: Multimodal Text Image Generation with Visual-harmony and Text-comprehension for Poster Design", in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, 7236–46.

[19] Y. Gao, Y. Guo, Z. Lian, Y. Tang, and J. Xiao, "Artistic glyph image synthesis via one-stage few-shot learning", *ACM Transactions on Graphics*, 2019.

[20] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks", in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2016, 2414–23.

[21] Y. Goda, T. Nakamura, and M. Kanoh, "Texture transfer based on continuous structure of texture patches for design of artistic Shodo fonts", *ACM SIGGRAPH ASIA 2010 Sketches*, 2010.

[22] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks", *Communications of the ACM*, 63(11), 2020, 139–44.

[23] S. Guo, Z. Jin, F. Sun, J. Li, Z. Li, Y. Shi, and N. Cao, "Vinci: an intelligent graphic design system for generating advertising posters", in *Proceedings of the 2021 CHI conference on human factors in computing systems*, 2021, 1–17.

[24] A. Gupta, J. Johnson, A. Alahi, and L. Fei-Fei, "Characterizing and Improving Stability in Neural Style Transfer", in *Proc. Int'l Conf. Computer Vision*, 2017, 4067–76.

[25] T. S. F. Haines, O. Mac Aodha, and G. J. Brostow, "My Text in Your Handwriting", *ACM Transactions on Graphics*, 35(3), May 2016, 26:1–26:18.

[26] J.-Y. He, Z.-Q. Cheng, C. Li, J. Sun, W. Xiang, X. Lin, X. Kang, Z. Jin, Y. Hu, B. Luo, *et al.*, "WordArt Designer: User-Driven Artistic Typography Synthesis using Large Language Models", *arXiv preprint arXiv:2310.18332*, 2023.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9), 2015, 1904–16.

[28] A. Hertzmann, "Paint by relaxation", in *Proc. Computer Graphics International*, IEEE, 2001, 47–54.

[29] A. Hertzmann, "Painterly rendering with curved brush strokes of multiple sizes", in *Prof. Conf. Computer graphics and interactive techniques*, 1998, 453–60.

[30] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin, "Image analogies", in *Proc. Siggraph*, 2001, 327–40.

[31] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models", *Advances in Neural Information Processing Systems*, 33, 2020, 6840–51.

[32] K. Hormann and G. Greiner, "MIPS: An efficient global parametrization method", *Curve and Surface Design: Saint-Malo 1999*, 2000, 153–62.

[33] Z. Hu, S. Liu, J. Jiang, R. Hong, M. Wang, and S. Yan, "PicWords: Render a picture by packing keywords", *IEEE transactions on multimedia*, 16(4), 2014, 1156–64.

[34] D. Huang, J. Li, C. Liu, and J. Liu, "AUPOD: end-to-end automatic poster design by self-supervision", *IEEE Access*, 10, 2022, 47348–60.

[35] H. Huang, H. Wang, W. Luo, L. Ma, W. Jiang, X. Zhu, Z. Li, and W. Liu, "Real-Time Neural Style Transfer for Videos", in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2017, 783–91.

[36] Q. Huang, B. Fu, A. Zhang, and Y. Qiao, "Gentext: Unsupervised artistic text generation via decoupled font and texture manipulation", *arXiv preprint arXiv:2207.09649*, 2022.

[37] Q. Huang, Q. Zhu, and S. Zhan, "Artistic Text Effect Transfer with Conditonal Generative Adversarial Network", in *2022 Asia Conference on Algorithms, Computing and Machine Learning (CACML)*, IEEE, 2022, 181–5.

[38] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz, "Multimodal unsupervised image-to-image translation", in *Proc. European Conf. Computer Vision*, 2018, 172–89.

[39] S. Iluz, Y. Vinker, A. Hertz, D. Berio, D. Cohen-Or, and A. Shamir, "Word-as-image for semantic typography", *ACM Transactions on Graphics*, 42(4), 2023, 1–11.

[40] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks", in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2017, 1125–34.

[41] J. Ji, G. Zhang, Z. Wang, B. Hou, Z. Zhang, B. Price, and S. Chang, "Improving diffusion models for scene text editing with dual encoders", *arXiv preprint arXiv:2304.05568*, 2023.

[42] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution", in *Proc. European Conf. Computer Vision*, 2016.

[43] P. Krishnan, R. Kovvuri, G. Pang, B. Vassilev, and T. Hassner, "Textstylebrush: transfer of text aesthetics from a single example", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[44] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks", *Communications of the ACM*, 60(6), 2017, 84–90.

[45] S. Lakhanpal, S. Chopra, V. Jain, A. Chadha, and M. Luo, "Refining Text-to-Image Generation: Towards Accurate Training-Free Glyph-Enhanced Image Generation", *arXiv preprint arXiv:2403.16422*, 2024.

[46] H.-Y. Lee, H.-Y. Tseng, J.-B. Huang, M. Singh, and M.-H. Yang, "Diverse image-to-image translation via disentangled representations", in *Proc. European Conf. Computer Vision*, 2018, 35–51.

[47] J. C. Lee, J. Forlizzi, and S. E. Hudson, "The kinetic typography engine: an extensible system for animating expressive text", in *Proceedings of the 15th annual ACM symposium on User interface software and technology*, 2002, 81–90.

[48] J. Lee, S. Jun, J. Forlizzi, and S. E. Hudson, "Using kinetic typography to convey emotion in text-based interpersonal communication", in *Proceedings of the 6th conference on Designing Interactive systems*, 2006, 41–9.

[49] J. Lee, Y. Kim, S. Kim, M. Yim, S. Shin, G. Lee, and S. Park, "Rewritenet: Realistic scene text image generation via editing text in real-world image", *arXiv preprint arXiv:2107.11041*, 1, 2021.

[50] J. Li, T. Tang, W. X. Zhao, J.-Y. Nie, and J.-R. Wen, "Pre-trained language models for text generation: A survey", *ACM Computing Surveys*, 56(9), 2024, 1–39.

[51] W. Li, Y. He, Y. Qi, Z. Li, and Y. Tang, "FET-GAN: Font and Effect Transfer via K-shot Adaptive Instance Normalization", in *Proc. AAAI Conf. Artificial Intelligence*, 2020.

[52] X. Li, L. Wu, X. Chen, L. Meng, and X. Meng, "Dse-net: Artistic font image synthesis via disentangled style encoding", in *2022 IEEE International Conference on Multimedia and Expo (ICME)*, IEEE, 2022, 1–6.

[53] Z. Lian, B. Zhao, and J. Xiao, "Automatic Generation of Large-scale Handwriting Fonts via Style Learning", in *SIGGRAPH ASIA 2016 Technical Briefs*, ACM, 2016, 12:1–12:4.

[54] S. Liao and K. Arakawa, "Interactive Poster Design System for Movies with StyleGAN", in *2021 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, IEEE, 2021, 1–2.

[55] J. Lin, M. Zhou, Y. Ma, Y. Gao, C. Fei, Y. Chen, Z. Yu, and T. Ge, "AutoPoster: A Highly Automatic and Content-aware Design System for Advertising Poster Generation", in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, 1250–60.

[56] P. Litwinowicz, "Processing images and video for an impressionist effect", in *Prof. Conf. Computer graphics and interactive techniques*, 1997, 407–14.

[57] Z. Liu, Y. Meng, H. Ouyang, Y. Yu, B. Zhao, D. Cohen-Or, and H. Qu, "Dynamic Typography: Bringing Text to Life via Video Diffusion Prior", *arXiv e-prints*, 2024, arXiv–2404.

[58] T. Lüddecke and A. Ecker, "Image segmentation using text and image prompts", in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2022, 7086–96.

[59] C. Luo, L. Jin, and J. Chen, "Siman: Exploring self-supervised representation learning of scene text via similarity-aware normalization", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, 1039–48.

[60] C. Luo, Q. Lin, Y. Liu, L. Jin, and C. Shen, "Separating content from style using adversarial learning for recognizing text in the wild", *International Journal of Computer Vision*, 129, 2021, 960–76.

[61] Y. Ma, F. Tang, W. Dong, and C. Xu, "Text Style Transfer With Decorative Elements", *2021 IEEE 4th International Conference on Multimedia Information Processing and Retrieval (MIPR)*, 2021, 330–6.

[62] R. Maharik, M. Bessmeltsev, A. Sheffer, A. Shamir, and N. Carr, "Digital Micrography", *ACM Transactions on Graphics*, 2011, 100:1–100:12.

[63] W. Mao, S. Yang, H. Shi, J. Liu, and Z. Wang, "Intelligent Typography: Artistic Text Style Transfer for Complex Texture and Structure", *IEEE Transactions on Multimedia*, 25, 2023, 6485–98.

[64] Y. Men, Z. Lian, Y. Tang, and J. Xiao, "A Common Framework for Interactive Texture Transfer", in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2018.

[65] Y. Men, Z. Lian, Y. Tang, and J. Xiao, "DynTypo: Example-based Dynamic Text Effects Transfer", in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2019.

[66] M. Minakuchi and K. Tanaka, "Automatic kinetic typography composer", in *ACM International Conference Proceeding Series*, Vol. 265, 2005, 221–4.

[67] S. Park, S. Chun, J. Cha, B. Lee, and H. Shim, "Few-shot font generation with localized style representations and factorization", in *Proc. AAAI Conf. Artificial Intelligence*, Vol. 35, No. 3, 2021, 2393–402.

[68] K. Peong, S. Uchida, and D. Haraguchi, "Typographic Text Generation with Off-the-Shelf Diffusion Model", *arXiv preprint arXiv:2402.14314*, 2024.

[69] B. Poole, A. Jain, J. T. Barron, and B. Mildenhall, "Dreamfusion: Text-to-3d using 2d diffusion", *arXiv preprint arXiv:2209.14988*, 2022.

[70] G. Pu, S. Xu, X. Cao, and Z. Lian, "Dynamic Texture Transfer using PatchMatch and Transformers", *arXiv preprint arXiv:2402.00606*, 2024.

[71] Y. Qu, Q. Tan, H. Xie, J. Xu, Y. Wang, and Y. Zhang, "Exploring stroke-level modifications for scene text editing", in *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37, No. 2, 2023, 2119–27.

[72] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision", in *Proc. IEEE Int'l Conf. Machine Learning*, PMLR, 2021, 8748–63.

[73] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks", in *Proc. Int'l Conf. Learning Representations*, 2016.

[74] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents. arXiv 2022", *arXiv preprint arXiv:2204.06125*, 2022.

[75] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever, "Zero-shot text-to-image generation", in *International conference on machine learning*, Pmlr, 2021, 8821–31.

[76] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models", in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2022, 10684–95.

[77] A. Rosenberger, D. Cohen-Or, and D. Lischinski, "Layered shape synthesis: automatic generation of control maps for non-stationary textures", *ACM Transactions on Graphics*, 28(5), 2009, 107.

[78] P. Roy, S. Bhattacharya, S. Ghosh, and U. Pal, "STEFANN: scene text editor using font adaptive neural network", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, 13228–37.

[79] A. Royer, K. Bousmalis, S. Gouws, F. Bertsch, I. Mosseri, F. Cole, and K. Murphy, "Xgan: Unsupervised image-to-image translation for many-to-many mappings", *Domain Adaptation for Visual Understanding*, 2020, 33–49.

[80] M. Ruder, A. Dosovitskiy, and T. Brox, "Artistic Style Transfer for Videos", in *German Conference on Pattern Recognition*, Springer, 2016, 26–36.

[81] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans, *et al.*, "Photorealistic text-to-image diffusion models with deep language understanding", *Advances in neural information processing systems*, 35, 2022, 36479–94.

[82] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans", in *Advances in Neural Information Processing Systems*, Vol. 29, 2016.

[83] J. Santoso, C. Simon, *et al.*, "On Manipulating Scene Text in the Wild with Diffusion Models", in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, 5202–11.

[84] W. Shimoda, D. Haraguchi, S. Uchida, and K. Yamaguchi, "De-rendering stylized texts", in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, 1076–85.

[85] W. Shimoda, D. Haraguchi, S. Uchida, and K. Yamaguchi, "Towards Diverse and Consistent Typography Generation", in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, 7296–305.

[86] P. Shivakumara, A. Banerjee, U. Pal, L. Nandanwar, T. Lu, and C.-L. Liu, "A new language-independent deep CNN for scene text detection and style transfer in social media images", *IEEE Transactions on Image Processing*, 2023.

[87] T. Su, F. Yang, X. Zhou, D. Di, Z. Wang, and S. Li, "Scene Style Text Editing", *arXiv preprint arXiv:2304.10097*, 2023.

[88] J. Subramanian, V. Chordia, E. Bart, S. Fang, K. Guan, R. Bala, *et al.*, "Strive: Scene text replacement in videos", in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, 14549–58.

[89] M. Tanveer, Y. Wang, A. Mahdavi-Amiri, and H. Zhang, "Ds-fusion: Artistic typography via discriminated and stylized diffusion", in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, 374–84.

[90] P. Tendulkar, K. Krishna, R. R. Selvaraju, and D. Parikh, "Trick or treat: Thematic reinforcement for artistic typography", *arXiv preprint arXiv:1903.07820*, 2019.

[91] Y. Tuo, W. Xiang, J.-Y. He, Y. Geng, and X. Xie, "AnyText: Multilingual Visual Text Generation And Editing", *arXiv preprint arXiv:2311.03054*, 2023.

[92] A. Van Den Oord, O. Vinyals, *et al.*, "Neural discrete representation learning", in *Advances in Neural Information Processing Systems*, Vol. 30, 2017.

[93] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need", in *Advances in Neural Information Processing Systems*, Vol. 30, 2017.

[94] P. Vincent, "A connection between score matching and denoising autoencoders", *Neural computation*, 23(7), 2011, 1661–74.

[95] C. Wang, L. Wu, X. Chen, X. Li, L. Meng, and X. Meng, "Letter Embedding Guidance Diffusion Model for Scene Text Editing", in *2023 IEEE International Conference on Multimedia and Expo (ICME)*, IEEE, 2023, 588–93.

[96] C. Wang, L. Wu, X. Liu, X. Li, L. Meng, and X. Meng, "Anything to glyph: Artistic font synthesis via text-to-image diffusion model", in *SIGGRAPH Asia 2023 Conference Papers*, 2023, 1–11.

[97] C. Wang, M. Zhou, T. Ge, Y. Jiang, H. Bao, and W. Xu, "Cf-font: Content fusion for few-shot font generation", in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2023, 1858–67.

[98] W. Wang, J. Liu, S. Yang, and Z. Guo, "Typography with Decor: Intelligent Text Style Transfer", in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2019.

[99] W. Wang, J. Xu, L. Zhang, Y. Wang, and J. Liu, "Consistent Video Style Transfer via Compound Regularization", in *Proc. AAAI Conf. Artificial Intelligence*, 2020, 1–8.

[100] Y. Wang, G. Pu, W. Luo, Y. Wang, P. Xiong, H. Kang, and Z. Lian, "Aesthetic text logo synthesis via content-aware layout inferring", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, 2436–45.

[101] Y. Wexler, E. Shechtman, and M. Irani, "Space-Time Completion of Video.", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3), March 2007, 463–76.

[102] L. Wu, C. Zhang, J. Liu, J. Han, J. Liu, E. Ding, and X. Bai, "Editing text in the wild", in *Proceedings of the 27th ACM international conference on multimedia*, 2019, 1500–8.

[103] S. Xiao, L. Wang, X. Ma, and W. Zeng, "TypeDance: Creating semantic typographic logos from image through personalized generation", in *Proceedings of the CHI Conference on Human Factors in Computing Systems*, 2024, 1–18.

[104] L. Xie, X. Shu, J. C. Su, Y. Wang, S. Chen, and H. Qu, "Creating emordle: Animating word cloud for emotion expression", *IEEE Transactions on Visualization and Computer Graphics*, 2023.

[105] L. Xie, Z. Zhou, K. Yu, Y. Wang, H. Qu, and S. Chen, "Wakey-Wakey: Animate Text by Mimicking Characters in a GIF", in *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, 2023, 1–14.

[106] Y. Xie, X. Chen, L. Sun, and Y. Lu, "Dg-font: Deformable generative networks for unsupervised font generation", in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2021, 5130–40.

[107] J. Xu and C. S. Kaplan, "Calligraphic packing", in *Proceedings of Graphics Interface 2007*, 2007, 43–50.

[108] L. Xu, Q. Yan, Y. Xia, and J. Jia, "Structure extraction from texture via relative total variation", *ACM Transactions on Graphics*, 31(6), 2012, 139.

[109] S. Xu, F. C. Lau, W. K. Cheung, and Y. Pan, "Automatic generation of artistic Chinese calligraphy", *IEEE Intelligent Systems*, 20(3), 2005, 32–9.

[110] X. Xu, L. Zhang, and T.-T. Wong, "Structure-based ASCII Art", *ACM Transactions on Graphics*, 29(4), July 2010, 52:1–52:9.

[111] M. Xue, Y. Ito, and K. Nakano, "An Art Font Generation Technique using Pix2Pix-based Networks", *Bulletin of Networking, Computing, Systems, and Software*, 12(1), 2023, 6–12.

[112] K. Yan, S. Yang, W. Wang, and J. Liu, "Multitask Attentive Network For Text Effects Quality Assessment", in *Proc. IEEE Int'l Conf. Multimedia and Expo*, 2020.

[113] Q. Yang, J. Huang, and W. Lin, "Swaptext: Image based texts transfer in scenes", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, 14700–9.

[114] S. Yang, J. Liu, Z. Lian, and Z. Guo, "Awesome typography: Statistics-Based Text Effects Transfer", in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2017.

[115] S. Yang, J. Liu, W. Wang, and Z. Guo, "TET-GAN: Text Effects Transfer via stylization and Destylization", in *Proc. AAAI Conf. Artificial Intelligence*, 2019.

[116] S. Yang, J. Liu, W. Yang, and Z. Guo, "Context-aware text-based binary image stylization and synthesis", *IEEE Transactions on Image Processing*, 28(2), 2018, 952–64.

[117] S. Yang, J. Liu, W. Yang, and Z. Guo, "Context-Aware Unsupervised Text Stylization", in *Proc. ACM Int'l Conf. Multimedia*, 2018.

[118] S. Yang, W. Wang, and J. Liu, "TE141K: Artistic Text Benchmark for Text Effect Transfer", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[119] S. Yang, Z. Wang, and J. Liu, "Shape-Matching GAN++: Scale Controllable Dynamic Artistic Text Style Transfer", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[120] S. Yang, Z. Wang, Z. Wang, N. Xu, J. Liu, and Z. Guo, "Controllable Artistic Text Style Transfer via Shape-Matching GAN", in *Proc. Int'l Conf. Computer Vision*, 2019.

[121] Y. Yang, D. Gui, Y. Yuan, W. Liang, H. Ding, H. Hu, and K. Chen, "GlyphControl: Glyph Conditional Control for Visual Text Generation", *Advances in Neural Information Processing Systems*, 36, 2024.

[122] B. Yu, Y. Xu, Y. Huang, S. Yang, and J. Liu, "Mask-guided GAN for robust text editing in the scene", *Neurocomputing*, 441, 2021, 192–201.

[123] Y. Yuan, Y. Ito, and K. Nakano, "Art font image generation with conditional generative adversarial networks", in *2020 Eighth International Symposium on Computing and Networking Workshops (CANDARW)*, IEEE, 2020, 151–6.

[124] F. Zhang, Y. Yang, W. Huang, G. Zhang, and J. Wang, "Improving font effect generation based on pyramid style feature", *International Journal of Performability Engineering*, 16(8), 2020, 1271.

[125] J. Zhang, Y. Wang, W. Xiao, and Z. Luo, "Synthesizing ornamental typefaces", in *Computer Graphics Forum*, Vol. 36, No. 1, Wiley Online Library, 2017, 64–75.

[126] J. Zhang, Z. Yang, L. Jin, Z. Lu, and J. Yu, "Creating Word Paintings Jointly Considering Semantics, Attention, and Aesthetics", *ACM Transactions on Applied Perceptions*, 19(3), 2022, 1–21.

[127] L. Zhang, X. Chen, Y. Wang, Y. Lu, and Y. Qiao, "Brush your text: Synthesize any scene text on images via diffusion model", in *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38, No. 7, 2024, 7215–23.

[128] L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text-to-image diffusion models", in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, 3836–47.

[129] Q. Zhang, S. Xiong, and A. Zhu, "Neural Style Transfer for Characters Synthesis via Stacked Network", *Aust. J. Intell. Inf. Process. Syst.*, 16(2), 2019, 50–8.

[130] Y. Zhang, Y. Zhang, and W. Cai, "Separating style and content for generalized style transfer", in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2018, 8447–55.

[131] Y. Zhao and Z. Lian, "UDiffText: A Unified Framework for High-quality Text Synthesis in Arbitrary Images via Character-aware Diffusion Models", *arXiv preprint arXiv:2312.04884*, 2023.

[132] A. Zhu, X. Lu, X. Bai, S. Uchida, B. K. Iwana, and S. Xiong, "Few-shot text style transfer via deep feature similarity", *IEEE Transactions on Image Processing*, 29, 2020, 6932–46.

[133] A. Zhu, Z. Yin, B. K. Iwana, X. Zhou, and S. Xiong, "Text style transfer based on multi-factor disentanglement and mixture", in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, 2430–40.

[134] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks", in *Proc. Int'l Conf. Computer Vision*, 2017, 2223–32.

[135] C. Zou, J. Cao, W. Ranaweera, I. Alhashim, P. Tan, A. Sheffer, and H. Zhang, "Legible compact calligrams", *ACM Transactions on Graphics*, 35(4), 2016, 1–12.