Font-Agent: Enhancing Font Understanding with Large Language Models

Yingxin Lai[1,2,†], Cuijie Xu[2], Haitian Shi[2], Guoqing Yang[1],
Xiaoning Li[1], Zhiming Luo[1,3,*], Shaozi Li[1,3]

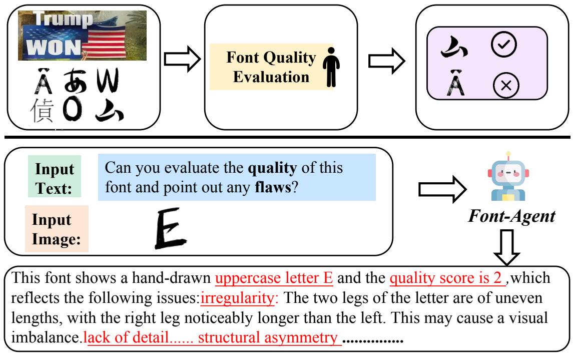
[1]Department of Artificial Intelligence, Xiamen University, Xiamen, China.
[2]Graph Origin.
[3]Key Laboratory of Multimedia Trusted Perception and Efficient Computing,
Ministry of Education of China, Xiamen University.

Abstract

The rapid development of generative models has significantly advanced font generation. However, limited exploration has been devoted to the evaluation and interpretability of graphical fonts. Existing quality assessment models can only provide basic visual analyses, such as recognizing clarity and brightness, without in-depth explanations. To address these limitations, we first constructed a large-scale multimodal dataset named the Diversity Font Dataset (DFD), comprising 135,000 font-text pairs. This dataset encompasses a wide range of generated font types and annotations, including language descriptions and quality assessments, thus providing a robust foundation for training and evaluating font analysis models. Based on this dataset, we developed a font agent built upon a Vision-Language Model (VLM) aiming to enhance font quality assessment and offer interpretable question-answering capabilities. Alongside the original visual encoder in VLM, we integrated an Edge-Aware Traces (EAT) module to capture detailed edge information of font strokes and components. Furthermore, we introduced a Dynamic Direct Preference Optimization (D-DPO) strategy to facilitate efficient model fine-tuning. Experimental results demonstrate that Font-Agent achieves state-of-the-art performance on the established dataset. To further evaluate the generalization ability of our algorithm, we conducted additional experiments on several public datasets. The results highlight the notable advantage of Font-Agent in both assessing the quality of generated fonts and comprehending their content.

1. Introduction

Font design has attracted significant attention because of its substantial commercial and academic importance, especially in fields like advertising and social media. Traditional font generation, which demands high precision, typically relies on manual design processes. However, the emergence of text-to-image generation models such as Midjourney [15], which have become milestones in the digital art and creative industries, has significantly impacted this field. This advancement enables the generation of personalized content, fosters immense creativity and innovation, and has led to a gradual reduction in design costs [20, 35]. Meanwhile, there is a growing demand for analyzing the huge amount of fonts generated in terms of correctness and quality.



Can you evaluate the **quality** of this font and point out any **flaws**?

This font shows a hand-drawn uppercase letter E and the quality score is 2 ,which reflects the following issues:irregularity: The two legs of the letter are of uneven lengths, with the right leg noticeably longer than the left. This may cause a visual imbalance.lack of detail...... structural asymmetry ....

Figure 1. Comparative Visualization of Font Quality Assessment: Our Automated Framework vs. Expert Select Approaches.

For the font quality assessment task, various classification models [6, 14, 54] have been proposed to produce probability values that indicate whether an input image is of good quality. Despite showing improvements in generalization, these models cannot provide convincing explanations for their assessments. In contrast, certain studies [45, 68] have attempted to use multimodal large language models (MLLMs) for font evaluation. However, font evaluation requires significant professional experience, an aspect not sufficiently incorporated into the training of MLLMs for font assessment,

[*]Corresponding Author: zhiming.luo@xmu.edu.cn
[†]This work was done during his internship at Graph Origin.

leading to sub-par performance. Thus, building high-level font quality assessment models still presents significant challenges. Firstly, there is a lack of high-quality font assessment datasets containing both quality labels and corresponding explanations. Secondly, there remains the challenge of effectively building feature representations between font images and textual descriptions.

To facilitate research in this area, we first construct a new multilingual font evaluation dataset comprising 135,000 manually annotated fonts across diverse languages and styles.This dataset is diverse, high-quality, and manually generated, covering multiple model outputs and languages such as English and Chinese. It includes detailed annotations, such as quality labels, mean opinion scores (MOS), and semantic tags, all carefully labeled by experts to ensure quality generation. This makes the dataset a strong foundation for future research in font generation and quality evaluation.

Building upon this dataset, we develop the **Font-Agent**, a novel font understanding framework based on the Vision-Language Model (VLM). The visual and textual encoders in the VLM first encode the font images and text questions into tokens. The decoder then generates the corresponding descriptions related to font quality assessments. Although VLM provides rich cross-domain semantic representations of font images and textual queries, we found it still struggles with understanding local curvature and stroke details essential for the semantic interpretation of font images. To address this issue, we propose an **Edge-Aware Traces (EAT)** module to capture detailed edge information. In our EAT module, we use a frequency-domain attention mechanism to enhance visual features related to high-frequency components in the font image. Then, the output feature of the EAT module is fused with the visual features from the VLM. Furthermore, we introduce a **Dynamic Direct Preference Optimization (D-DPO)** strategy for training Font-Agent. D-DPO addresses key limitations of traditional methods by optimizing dense, fine-grained, segment-level preferences, ensuring that hallucinated segments receive stronger feedback and maintaining factual accuracy. Consequently, D-DPO not only improves the model's learning efficiency but also mitigates the issue of uniformly weighting all words, enhancing the model's capability to distinguish relevant from irrelevant content. D-DPO dynamically adjusts each character's importance based on uncertainty, consistency, and font features, ensuring stronger feedback for high-quality characters while minimizing the impact of lower-quality ones. Finally, we conduct experiments on our constructed DFD dataset and other widely used benchmark datasets to evaluate the effectiveness of our proposed method.

This paper's contributions are summarized as follows:

- **Large-Scale Multilingual Font Dataset:** We contrust a high-quality multilingual font evaluation dataset comprising 135,000 manually annotated fonts across diverse languages and styles, addressing real-world font evaluation challenges.
- **Multimodal Font Representation:** We propose **Font-Agent**, a method that leverages visual-language models with multimodal evaluation capabilities. By applying convolutional layers to the phase and amplitude spectra of font images, Font-Agent excels at assessing fine-grained font details for quality evaluation and font-related question-answering tasks.
- **State-of-the-Art Performance:** Font-Agent achieves leading performance in font quality assessment. Our analyses demonstrate its effectiveness and generalization in font evaluation.

2. Related Work

2.1. Recent Studies on Font Generation

Recently, various approaches have significantly advanced font generation. For example, transformer-based methods, such as DeepVecFont [62], enhance diversity but are limited by a small set of glyphs. GAN-based methods, such as FUNIT [34], achieve style transfer but often suffer quality issues when generating complex fonts. Unsupervised approaches, such as DG-Font [71], show promising visual results, but require extensive data alignment. To capture local style details, methods such as LF-Font [42] and MX-Font [41] utilize font component segmentation, while CF-Font [60] incorporates style vectors to improve consistency across fonts. The diffusion model FontDiffuser [70] introduces contrastive learning to address complex font synthesis but requires extensive pretraining. FS-Font [55] aligns the spatial correspondence between content and style, but is highly sensitive to the quality of reference fonts. Despite significant progress in font generation, a key challenge remains: an effective evaluation of font quality. Existing methods primarily focus on improving generation quality, lacking robust and interpretable frameworks for assessing fine-grained font quality, which is crucial for real-world applications requiring high visual consistency and style accuracy.

2.2. Image Quality Assesement

Image quality assessment (IQA) is a crucial task in computer vision that directly impacts the performance of image generation systems. Common evaluation models, such as Xception [48], Meso4 [1], and EfficientNet-B4 [54], primarily focus on evaluating overall image similarity or detecting anomalies in natural images. However, with the rise of multimodal text-image models, evaluation approaches have shifted towards measuring text-image similarity [46]. For example, ImageReward [67] and PickScore [25] enhance robustness by fine-tuning vision-language models (VLM) using large datasets with human-annotated scores. Although these methods perform well to assess natural image quality,

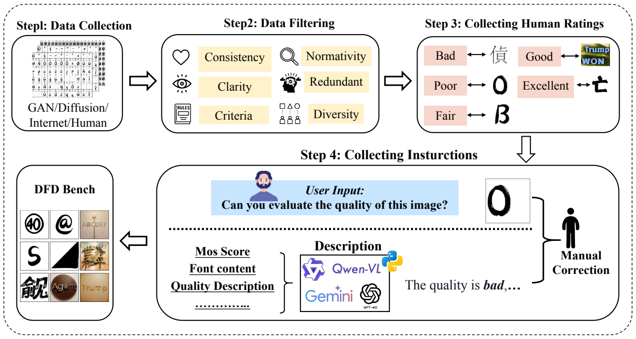


Figure 2. The collection process for DFD Benchmark.

they face significant challenges when applied to symbolic graphics, particularly font images [39]. Current models often overlook fine-grained font details, such as stroke thickness, character precision, and style consistency—critical elements in evaluating font-generated images. Moreover, as noted in [45], large language models (LLMs) lack the ability to comprehend local curvature and stroke details essential for the semantic interpretation of font images. This limitation highlights the difficulty these models face in capturing nuanced visual characteristics of fonts, which differ structurally and semantically from natural images.Therefore, designing specialized models for font evaluation is crucial. Unlike general IQA models, font evaluation requires detailed analyses of local features, focusing on detail preservation, style consistency, and readability. Existing models, which mainly assess global image features, struggle with the unique challenges presented by fonts. A dedicated framework for font quality assessment could effectively address these issues, offering improved evaluation accuracy, and interpretability.

3. Method

3.1. DFD Benchmark

In this section, we introduce the DFD dataset for assessing font quality, as shown in Fig. 2 . The dataset construction process, including collection, filtering and annotation.

Font Data Collection: In this study, we collected font data from a variety of platforms to ensure diversity and compliance with licensing requirements. Data sources included international resources such as Google Fonts, Font Squirrel , DaFont , and 1001 Free Fonts, as well as local Chinese-focused platforms such as Fangzheng, Hanyi, and Zihun. To enhance data diversity, we collected approximately 63% English fonts, 26% Chinese fonts, and 11% other language fonts. English fonts generally feature simpler stroke structures, whereas Chinese fonts exhibit more complex compositions, which highlights the structural diversity inherent in font design. In terms of data composition, about 20% of the fonts were manually created, while 80% was generated using models such as GANs and Stable Diffusion. Specifically,we employed most common font generation models—including VQ-Font [72] , DG-Font [71], and diffusion-based models like FontDiffuser [70] and Diff-Font [12] to ensure sample diversity across our dataset.Similarly we have generated a colorful logo design model Design, using 9 common models including Diff-Text [76] , Anything to Glyph [59], WordArt Designer [66], DS-Fusion [56], DynTypo [37], DynTexture [44], SwapText [69], those cover most of the scenarios of daily font design.

Quality Filtering: In this stage, we manually filter low-resolution fonts, structural errors, and unbalanced proportions. We also removed fonts with similar styles and language features to ensure diversity.After filtering, the dataset contains a total of 135,000 high-quality font samples suitable for quality evaluation tasks.

Semantic Labeling: In the semantic labeling process, a team of 15 professional annotators, also experts in the font industry, meticulously labeled various aspects of the fonts. Each font was assigned a quality label (high, medium, or low) and a Mean Opinion Score (MOS) ranging from 1 to 5. The annotations included the specific text displayed and the font language, along with potential applications such as casual, business, educational, or artistic uses. The fonts were categorized into styles such as serif, sans serif, handwritten, and decorative. Justifications for the quality ratings were provided, emphasizing readability, aesthetic appeal, technical correctness, and suitability for intended use. To further refine font feature descriptions, advanced models such as GPT-4o, Gemini-Pro, and QwenVL were employed for automated description generation, with manual corrections made to address inaccuracies. As a result, the DFD dataset includes 42,362 pairs of English instruction-image samples derived from 5,321 SVG files, 31,544 pairs of Chinese instruction-image samples from 3,609 SVG files, 27,690 character samples - including digits and geometric shapes from 5,139 SVG files, and 33,404 font logo design instruction-image samples from 39,812 SVG files. This comprehensive dataset provides a solid foundation for future research in font generation and quality assessment. Regarding the composition of the dataset, approximately 63% are English fonts (approximately 85,050 samples), 26% are Chinese fonts (about 35,100 samples), and 11% are fonts in other languages (about 14,850 samples). In terms of generation methods, 80% of the fonts are machine generated (approximately 108, 000 samples), while 20% are manually crafted (approximately 27,000 samples). The quality level distribution is as follows: high quality (MOS 4-5) includes 54,000 samples, accounting for 40% of the dataset; medium quality (MOS 2-3) includes 67,500 samples, accounting for 50%; and low quality (MOS 1) includes 13,500 samples, accounting for 10%.

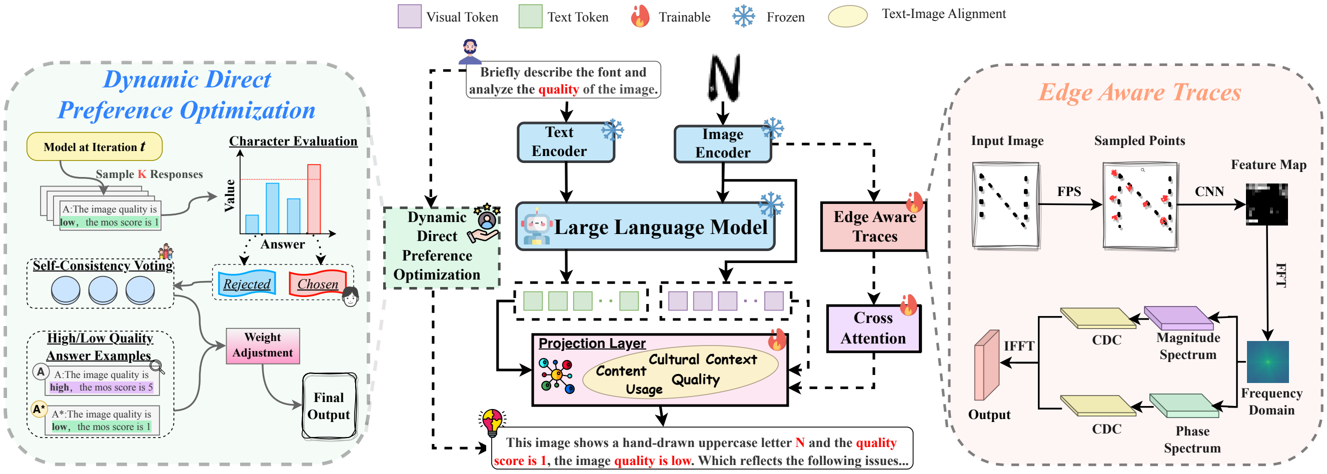

Visual Token
Text Token
Trainable
Frozen
Text-Image Alignment
Dynamic Direct Preference Optimization
Model at Iteration t
Sample K Responses
A:The image quality is low, the mos score is 1
Character Evaluation
Value
Answer
Self-Consistency Voting
Rejected
Chosen
High/Low Quality Answer Examples
A
A:The image quality is high, the mos score is 5
A*
A*:The image quality is low, the mos score is 1
Weight Adjustment
Final Output
Briefly describe the font and analyze the quality of the image.
Text Encoder
Image Encoder
Large Language Model
Dynamic Direct Preference Optimization
Edge Aware Traces
Cross Attention
Projection Layer
Cultural Context
Content
Quality
Usage
This image shows a hand-drawn uppercase letter N and the quality score is 1, the image quality is low. Which reflects the following issues...
Edge Aware Traces
Input Image
Sampled Points
FPS
CNN
Feature Map
FFT
Frequency Domain
Magnitude Spectrum
Phase Spectrum
CDC
CDC
IFFT
Output

Figure 3. Multi-Modal Font network (Font-Agent) architecture.

3.2. Proposed Font-Agent

Multi Model Representation. We introduce Font-Agent, a multi-modal font evaluation method based on the fine-tuned Qwen-VL model [2]. Given an input image X, the font image is processed through a visual encoder E_V to extract visual representation tokens F_v:

$$F_v = \{n_1, n_2, \ldots, n_N\} \in \mathbb{R}^{n \times D} \tag{1}$$

Where D is the dimensionality of each token,these tokens capture rich semantic information and demonstrate strong generalization for font recognition tasks. A text instruction t is sampled from a predefined template Q to guide the LLM in font recognition, represented by text tokens F_t:

$$F_t = \{m_1, m_2, \ldots, m_M\} \in \mathbb{R}^{m \times D} \tag{2}$$

This instruction helps the pretrained model M_t understand the visual content F_v and extract relevant features, leveraging the capabilities of Qwen-VL [2]. Our method differs from previous work by integrating both visual and textual encoders, rather than relying solely on the visual encoder. The multimodal representation F_T is first obtained by combining both F_v and F_t. Then, the final multimodal representation for font analysis is computed by first projecting F_T through a linear layer $proj$ and subsequently concatenating the result with F_v.

$$F = \text{Cat}\left([\text{proj}(F_T), \text{proj}(F_v)]\right), F_T = E_L(t, F_v) \tag{3}$$

Edge Aware Traces. Capturing the fine geometric details of glyphs is crucial for font recognition and quality assessment. We extract control points and points along the Bézier curves defined by each glyph, forming a set of points $P = \{p_i \in \mathbb{R}^2 \mid i = 1, 2, \ldots, M\}$. To efficiently represent the shape of the glyph and reduce computational complexity, we apply Farthest Point Sampling (FPS) to select a subset $S = \{p_0, p_1, \ldots, p_{N-1}\}$ as representative points. Using these sampled points, we construct a saliency map $W(x, y)$, by setting $W(x_k, y_k) = 1$ for each point $p_k \in S$. Then, we apply a Gaussian filter to smooth the map, highlighting the significant regions of the glyph. Next, we pass the input glyph image X through a Convolutional Neural Network (CNN) to obtain the feature map $F(x, y) = \text{CNN}(X)$.

We then transform both the feature map and the saliency map into the frequency domain using the Fast Fourier Transform(FFT):

$$F(u, v) = \mathcal{FFT}\{F(x, y)\}, \quad W(u, v) = \mathcal{FFT}\{W(x, y)\} \tag{1}$$

where (u, v) represents the frequency indices. To emphasize the frequency components corresponding to the important spatial regions, we define an adaptive weighting function:

$$H(u, v) = 1 + \gamma \cdot \frac{|W(u, v)|}{\max_{u,v} |W(u, v)|} \tag{2}$$

where γ is a scaling factor controlling the emphasis strength. We enhance the frequency domain feature map by element-wise multiplication F_{weighted},then, we apply the inverse Fourier transform(IFFT) to convert it back to the spatial domain, obtaining the enhanced feature map:

$$\begin{aligned} F_{\text{weighted}}(u, v) &= F(u, v) \cdot H(u, v) \\ F_{\text{enhanced}}(x, y) &= \mathcal{F}^{-1}\{F_{\text{weighted}}(u, v)\}. \end{aligned} \tag{3}$$

To further capture fine details crucial for distinguishing subtle font differences, we enhance the high-frequency components. We apply a high-pass filter:

$$H_{\text{high}}(u, v) = 1 - \exp\left(-\frac{u^2 + v^2}{2\sigma^2}\right). \tag{4}$$

In order to isolate high-frequency features, where σ controls the bandwidth of the filter. The filtered frequency components are:

$$F_{\text{high}}(u,v) = F(u,v) \cdot H_{\text{high}}(u,v) \tag{5}$$

We extract the magnitude and phase:

$$\gamma(u,v) = |F_{\text{high}}(u,v)|, \quad \phi(u,v) = \arg\{F_{\text{high}}(u,v)\}. \tag{6}$$

Then, apply convolution to enhance them:

$$\gamma_{\text{cdc}} = Conv_{\text{cdc}}(\gamma), \quad \phi_{\text{cdc}} = Conv_{\text{conv}}(\phi), \tag{7}$$

where $Conv_{\text{cdc}}$ denotes the Central Difference Convolutional (CDC) operator [75]. We adopt CDC to integrate local details into MLLMs and to explore fine-grained difference information from neighboring pixels. We reconstruct the enhanced frequency components $F_{\text{enhanced}}(u,v)$ and transform them back to the spatial domain:

$$\begin{aligned} F_{\text{enhanced}}(u,v) &= \gamma_{\text{cdc}}(u,v) \cdot e^{i\phi_{\text{cdc}}(u,v)} \\ F_{\text{high-enhanced}}(x,y) &= IFFT\{F_{\text{enhanced}}(u,v)\}. \end{aligned} \tag{8}$$

We further modulate the enhanced features using the saliency map to emphasize important regions:

$$F_{\text{final}}(x,y) = F_{\text{enhanced}}(x,y) \cdot (1 + \alpha \cdot W(x,y)), \tag{9}$$

where α is a modulation parameter controlling the influence of the saliency map $W(x,y)$. This integrated method enables our model to focus on both critical spatial regions and high-frequency components, effectively capturing the fine details of glyphs and improving performance in font recognition and quality assessment tasks.The combination of F_{final} and F is achieved using cross-attention, as referenced in [78].

D-DPO. We propose a novel dynamic adaptive preference optimization method for font quality assessment, which fully considers the unique characteristics of fonts to achieve fine-grained quality control. Unlike traditional methods that rely solely on log probabilities, we introduce an adaptive weighting strategy based on character uncertainty $Q(y_i)$, contextual consistency $C(y_i)$, glyph features $F(y_i)$, and self-consistency voting $V(y_i)$. This strategy dynamically adjusts the weight of each character in the quality assessment, prioritizing characters with high self-consistency and high font quality while reducing the impact of low quality characters. Our log probability computation formula is:

$$\log \pi(y \mid x) = \frac{1}{N} \sum_{y_i \in y} w(y_i) \log p(y_i \mid x, y_{<i}), \tag{10}$$

where y is the font character sequence to be evaluated, x represents the input data or context, $p(y_i \mid x, y_{<i})$ is the probability of character y_i given the input x and previous characters $y_{<i}$, and N is a normalization factor to ensure appropriate weight normalization. The adaptive weight function $w(y_i)$ is defined as:

$$w(y_i) = \left(\frac{k}{V(y_i)}\right) \times \begin{cases} \epsilon, & \text{if } Q(y_i) \geq \theta_{\text{high}}, \\ & C(y_i) \geq \theta_{\text{consistent}}, \\ & F(y_i) \geq \theta_{\text{font}}; \\ \alpha, & \text{otherwise.} \end{cases} \tag{11}$$

where $V(y_i)$ is the self-consistency voting count for character y_i, counting the occurrences of position i in the k candidate outputs $\{\hat{y}^1, \hat{y}^2, \ldots, \hat{y}^k\}$. Thresholds $\theta_{\text{high}}, \theta_{\text{consistent}}, \theta_{\text{font}}$ define high-quality characters based on uncertainty, contextual consistency, and font features. The bias parameter $\epsilon > 1$ amplifies the influence of high-quality characters, and α is a small positive number (typically $0 < \alpha < 1$) used to reduce the weight of low-quality characters. The normalization factor N is defined as:

$$N = \epsilon \sum_{y_i \in y_h} \frac{k}{V(y_i)} + \sum_{y_i \in y_l} w(y_i), \tag{12}$$

where y_h is the set of high-quality characters satisfying $Q(y_i) \geq \theta_{\text{high}}, C(y_i) \geq \theta_{\text{consistent}}, F(y_i) \geq \theta_{\text{font}}$, and y_l is the remaining set of characters that do not meet the high-quality standards. To evaluate font quality using adaptive weights, we define:

$$R_w = \frac{1}{N_w} \sum_{y_i \in y_w} w(y_i) \log \frac{p^*(y_i \mid x, y_{<i})}{p_{\text{ref}}(y_i \mid x, y_{<i})}, \tag{13}$$

$$R_l = \frac{1}{N_l} \sum_{y_i \in y_l} w(y_i) \log \frac{p^*(y_i \mid x, y_{<i})}{p_{\text{ref}}(y_i \mid x, y_{<i})}, \tag{14}$$

where $p^*(y_i \mid x, y_{<i})$ is the probability under the optimized strategy, $p_{\text{ref}}(y_i \mid x, y_{<i})$ is the probability under the reference strategy, and N_w, N_l are normalization factors. The loss function L_{ddpo} is defined as:

$$L_{\text{ddpo}} = -E(x, y_w, y_l) \left[\log \sigma(\beta(R_w - R_l))\right], \tag{15}$$

where σ is the sigmoid function, and β is a scaling parameter. Our method improves sensitivity to subtle variations that affect readability and appearance by focusing on the basic shapes of the font. The self-consistency voting $V(y_i)$ leverages the model's own predictions to improve reliability without adding extra computational cost.

4. Experiments

4.1. Datasets and Metrics.

Datasets. To validate the effectiveness of *Font-Agent*, we conduct extensive experiments on multiple datasets. For

Method	Detector	Backbone	DFD-UN		DFD-CN		DFD-NUM		Avg.	
			ACC (%)	AUC (%)	ACC (%)	AUC (%)	ACC (%)	AUC (%)	ACC (%)	AUC (%)
Naive	Meso4 [1]	MesoNet	76.3	74.5	77.2	75.3	76.8	74.9	76.8	74.9
Naive	MesoIncep [1]	MesoNet	78.5	76.7	78.4	73.8	78.3	76.1	78.4	75.53
Naive	CNN-Aug [13]	ResNet	82.6	78.2	78.2	79.5	79.8	77.4	79.7	78.3
Naive	Xception [48]	Xception	89.3	87.9	84.9	88.4	85.6	85.8	87.9	86.0
Naive	EfficientB4 [54]	EfficientNet	86.5	84.7	84.7	85.6	86.3	84.2	85.8	84.8
LLM	LLaVA-v1.6 [33]	Vicuna7B	63.2	60.5	60.5	60.5	63.1	60.9	62.6	60.6
LLM	mPLUG-Owl2 [73]	LLaMA-2 7B	73.2	71.5	71.5	74.2	73.6	71.8	72.8	72.5
LLM	InternVL2 [4]	LLaMA2 7B	71.5	69.3	69.3	72.4	71.8	69.6	70.5	70.8
LLM	GPT-4o [40]	-	62.1	58.3	61.4	63.7	64.0	59.6	62.5	60.5
LLM	BLIP2 [30]	OPT 7B	75.1	73.8	73.8	74.6	75.0	73.5	74.6	74.3
Ours	-	-	93.9(↑4.6)	96.8(↑8.9)	94.5(↑9.6)	91.8(↑3.4)	93.1(↑6.8)	96.7(↑10.9)	93.8(↑5.9)	95.1(↑9.1)

Table 1. Visual reasoning performance on the DFD dataset, evaluating the ability to understand and reason about English (DFD-UN), Chinese (DFD-CN), and numeric characters (DFD-NUM). The top model is marked in red, the second in blue, and improvements in green.

font quality assessment,we utilize our self-built DFD dataset along with six widely used public datasets: LIVE [49], CSIQ [26], KADID-10k [31], BID [7], CLIVE [10], and KonIQ-10k [16]. For the font recognition task , we evaluate *Font-Agent* on 12 common font recognition datasets: Regular benchmarks (IIIT5k [38], SVT [52], IC13 [21]), Irregular Benchmarks (IC15 [22], SVTP [43], CUTE80 [47]), COCO Text (COCO) [58], CTW [36], Total Text (TT) [5], OST excluded benchmarks (HOST and WOST) [63], and the artistic benchmark WordArt [66], and the DFD dataset.

Metrics. For the evaluation metrics, we employ Accuracy (ACC) and Area Under the Curve (AUC) as the primary measures to assess the model's ability to distinguish between high-quality and low-quality fonts. Additionally, we use Spearman's Rank Correlation Coefficient (SRCC) and Pearson's Linear Correlation Coefficient (PLCC) to evaluate the correlation between predicted scores and ground-truth rankings in font quality assessment. We also measure font recognition performance using Word Accuracy.

Implementation Details. To train our multimodal font quality assessment model, we followed the Qwen-VL [2] strategy. We utilized the AdamW optimizer with a weight decay of 0.01, a cosine learning rate scheduler, and no warm-up phase. The first training phase lasted 15 epochs with a batch size of 16 and an initial learning rate of 1×10^{-4}. In the second phase, the learning rate was reduced to 4×10^{-6} for an additional 8 epochs, while the batch size remained at 16. Training was conducted on four NVIDIA A100 80GB GPUs. Further details are provided in the Appendix.

4.2. Evaluation of Font Quality and Recognition Performance

We compare our method with four categories of font quality assessment methods: **(1) Traditional classification methods**, including Meso4 [1], CNN-Aug [13], Xception [48], and EfficientNet-B4 [54]; **(2) Multimodal large model methods**, such as LLaVA-v1.6 [33], mPLUG-Owl2 [73], InternVL2 [4], GPT-4o [40], and BLIP2 [30]; **(3) Quality assessment models**, including IDEFICS [18], XComposer-VL-2 [8], Co-Instruct [65], and Compare2Score [65]; **(4) Commonly used text recognition models**, including NIMA [53], MLSP [11], MUSIQ [23], MaxViT [57], CLIP-IQA+ [61], Aesthetic Predictor [17], LIQE [77], VILA [24], and Q-Align [64]. In this section, we follow the approach employed in Q-Align [64]. For methods not specifically designed for quality assessment, we retrain them either by mapping the quality labels of the data to binary values (0 and 1) or by using our font-text pairs. All methods are trained under our proposed protocols. We adopt the hyperparameters described in the corresponding papers or optimally select improved ones when necessary.

Protocol 1: Traditional Classification Models. We evaluated various classification models in cross-language scenarios using datasets designed to discriminate font quality. Annotated image quality labels were employed for classification label mapping and retraining. Our Font-Agent method achieved the best performance in font quality classification. As shown in Table 1, compared to the top-ranking Xception [6] method, Font-Agent exhibits an average improvement of over 5% in both ACC and AUC on the DFD dataset. Performance enhancement is even more significant when compared to widely used multimodal large models, such as GPT-4o [40] and BLIP2 [29]. These results indicate Font-Agent's effectiveness in overcoming unreliability and inefficiency issues in font recognition tasks.

Protocol 2: Multimodal Large Model Methods. We further evaluated common multimodal large models in font quality assessment. Since traditional methods lack domain-specific knowledge regarding "high" and "low" font quality, directly applying them for comparison is meaningless. Therefore, we provided our annotated font-text pairs as inputs to these methods across the datasets. As shown in Table 1 and Table 3, despite significant domain shifts from natural images to graphical images, which typically pose substantial challenges to most multimodal large models during domain adaptation, our proposed Font-Agent still achieves the best performance (see Table 3). This result highlights the effectiveness of our method in extracting detailed graphic features.

Protocol 3: Image Quality Assessment Models. As shown in Table 1, our Font-Agent outperforms other algorithms in the three cross-language testing scenarios. An interesting observation is that VILA [24] and Q-Align [64] are currently the most advanced image understanding models, and Q-Align [64] is specifically designed for image quality assessment. However, under the premise of graphical font input, they did not consider the issues of modality imbalance and unreliability brought about by font graphicalization. Specifically, some details may be ignored or distorted during the graphicalization process, affecting the accuracy and consistency of the assessment. This leads to differences of over 3% in average SRCC and PLCC compared to our method, further highlighting the effectiveness of our proposed EAT and D-DPO in improving precise control over details.

Protocol 4: Font Recognition Models. Compared to Protocol 1, the multi-font input exacerbates the impact of complex strokes, causing the models to underperform in font recognition tasks. However, in this sub-protocol comparison, we observe that our method still achieves the best performance across various evaluation metrics, even when tested on multiple public font recognition datasets under diverse and complex conditions (see Table 4). This highlights the advantages of our proposed multimodal model architecture, particularly for font recognition tasks. Specifically, while existing multimodal large models encounter significant challenges in font recognition and quality assessment, our model substantially improves recognition capabilities by effectively capturing detailed font features, demonstrating robust adaptability, especially in “complex graphical deployment environments.”

Training Set: **DFD$_{\text{train}}$**		→*Testing Set:*	**DFD$_{\text{test}}$**	
Method	#Training	Extra Data?	SRCC	PLCC
NIMA (TIP 2018) [53]	125K (92%)	✗	0.608	0.632
MLSP (CVPR 2019) [11]	125K (92%)	✗	0.650	0.655
MUSIQ (ICCV 2021) [23]	125K (92%)	✗	0.720	0.735
MaxViT (ECCV 2022) [57]	125K (92%)	✗	0.705	0.740
CLIP-IQA+ (AAAI 2023) [61]	125K (92%)	✗	0.715	0.718
Aesthetic Predictor (2023) [17]	125K (92%)	✗	0.718	0.720
LIQE (CVPR 2023) [77]	125K (92%)	✗	0.770	0.760
VILA (CVPR 2023) [24]	125K (92%)	✓	0.772	0.770
Q-Align (ICML 2024) [64]	125K (92%)	✗	0.785	0.815
Ours	125K (92%)	✗	**0.830**	**0.823**

Table 2. **Font-Agent** performance on font quality assessment (DFD). All methods are trained using the custom split setting of our self-established dataset.

4.3. Ablation Study and Discussion

Effectiveness of Edge Aware Traces. To validate the effectiveness of the proposed EAT module, we compare the network's performance before and after removing the module. Additionally, we integrate EAT into the backbone of various font-point sampling methods (i.e., Random Sampling, KNN Sampling, and FPS Sampling) and test different filter bandwidths to analyze performance differences between using Qwen-VL combined with EAT and using Qwen-VL alone. As shown in Table 5, methods equipped with additional frequency-domain detail extraction exhibit a clear performance improvement over the basic multimodal backbone network. This improvement is especially pronounced without the inclusion of other sampling methods. Furthermore, as illustrated in Fig. 4, when integrating our proposed FPS sampling method, the enhancement in detail extraction significantly surpasses that achieved with Random Sampling and KNN methods. These results demonstrate the effectiveness and general applicability of EAT for existing quality assessment and multimodal strategies.

Method	LIVE	CSIQ	KADID-10k	BID	CLIVE	KonIQ-10k
IDEFICS [18]	0.125	0.669	0.500	0.523	0.146	0.727
LLaVA-1.5 [32]	0.170	0.544	0.600	0.579	0.074	0.455
mPLUG-Owl2 [74]	0.484	0.394	0.302	0.613	0.407	0.273
XComposer-VL-2 [8]	0.045	0.662	0.672	0.648	0.067	0.059
Co-Instruct [65]	0.672	0.426	0.391	0.695	0.718	0.849
Compare2Score [65]	0.849	0.720	0.870	**0.861**	0.788	0.858
Ours	**0.873**	**0.741**	**0.885**	0.839	**0.807**	**0.873**

Table 3. Performance comparison in terms of prediction accuracy on six most common Image Quality Assessment datasets. The best results are highlighted in boldface.

Impact of the D-DPO. Here, our aim is to answer two questions: (1) Is solving the font understanding problem through D-DPO more appropriate compared to existing Direct Preference Optimization (DPO) methods ? (2) Is our D-DPO more capable of enhancing the performance of complex multimodal fusion frameworks like Qwen-VL+EAT compared to existing multimodal methods? From Table 5 , we can observe that when Qwen-VL+EAT is used as the backbone network, our D-DPO surpasses existing large model methods and traditional classification methods. Even for the vanilla Qwen-VL, our D-DPO can enhance its performance. This indicates the effectiveness of addressing the font quality assessment problem in complex scenarios by focusing on potential errors. The results in Fig.4 also indicate that, compared to existing state-of-the-art fine-tuning methods (such as DPDP, IPO and TDPO), our D-DPO is more suitable for Qwen-VL+EAT, although these three variants modulate during modality gradient conflicts and feature fusion, respectively. Although these two variants improve performance compared to not fine-tuning at all, their performance is inferior in all cases to that of D-DPO, which modulates in all situations. This validates the necessity of adaptive modulation in different scenarios.In Fig. 5, we show the accuracy rates and hallucination counts for models with different parameter sizes, using varying amounts of feedback data. On the DFD dataset, as feedback data increases, Font-Agent's hallucination rate and number of hallucinated segments decrease rapidly. This shows that detailed corrective feedback

Method	Venue	Regular			Irregular						Occluded		Others	DFD	AVG
		IIIT	SVT	IC13	IC15	SVTP	CT80	COCO	CTW	TT	HOST	WOST	WordArt	-	
		3000	647	1015	2077	645	288	9896	1572	2201	2416	2416	1511	-	
ASTER [51]	PAMI'18	95.03	89.49	93.79	85.48	82.02	90.28	62.25	76.53	78.69	43.34	64.65	65.59	56.34	75.65
NRTR [50]	ICDAR'19	97.43	93.82	96.06	85.15	84.03	91.32	65.94	81.74	81.83	50.83	71.52	64.06	61.03	78.82
SAR [28]	AAAI'19	97.70	94.13	96.35	87.47	87.60	93.06	67.41	83.91	86.23	46.36	70.32	72.40	55.96	79.91
SATRN [27]	AAAI'20	97.83	95.83	97.44	89.46	90.85	96.18	73.06	84.61	87.91	56.71	75.62	75.71	68.05	83.78
ABINet [9]	CVPR'21	97.90	95.98	96.16	91.66	90.23	93.75	71.46	83.72	86.01	56.54	75.75	75.25	71.54	83.53
PARSeq* [3]	ECCV'22	99.10	97.84	98.13	89.22	96.90	98.61	-	-	-	-	-	-	73.63	-
MAERec [19]	ICCV'23	98.93	97.99	98.62	93.04	94.57	98.96	78.84	88.87	93.91	73.97	85.72	82.59	77.08	89.46
E^{2}STR [78]	CVPR'24	99.10	98.15	98.03	92.99	96.43	98.96	77.29	88.36	93.46	73.30	85.51	81.47	76.67	89.20
Ours	-	99.42	98.71	99.14	94.06	97.85	99.59	78.60	91.36	96.83	76.54	87.70	84.49	83.57	91.32

Table 4. Performance comparison of font recognition models on a comprehensive benchmark based on the E^{2}STR [78] framework. Red and Blue indicate the best and second-best performances, respectively.

Method	DFD-UN		DFD-CN		DFD-NUM	
	ACC (%)	AUC (%)	ACC (%)	AUC (%)	ACC (%)	AUC (%)
Pre-trained MLLM	57.5	60.3	60.4	58.7	59.2	58.1
Instructed-MLLM	79.4	81.2	79.6	80.4	80.7	77.8
EAT only	83.5	84.2	84.0	85.0	83.8	84.5
D-DPO only	86.4	86.6	86.2	86.1	86.5	86.3
Ours	**93.6**	**96.8**	**94.5**	**91.8**	**93.1**	**96.7**

Table 5. Ablation study evaluating the effectiveness of each proposed module. The results demonstrate incremental improvements contributed by each component.

helps align the behavior of multi-modal large language models (MLLMs) effectively.

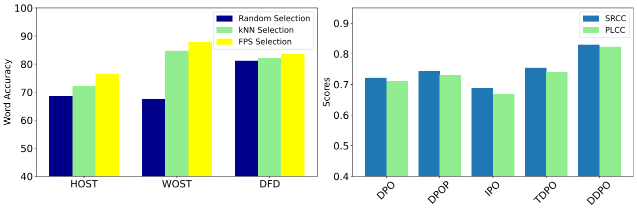


Figure 4. Qualitative results of the font quality assessment with different point selection and different dpo method.

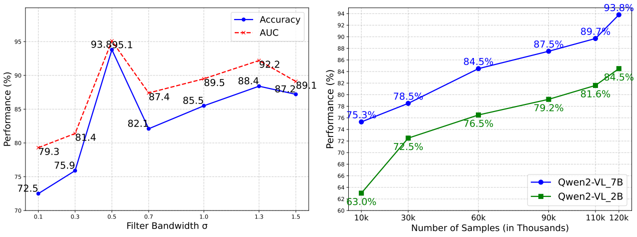
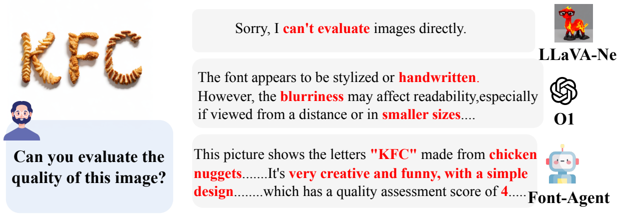


Figure 5. Comparison between different filter bandwidth strategies and scaling effects of annotation quality for quality assessments performance on DFD.

Discussion. Fig. 5 shows accuracy and hallucination rates for models with varying parameters and feedback data. On


Can you evaluate the quality of this image?

Sorry, I **can't evaluate** images directly.

The font appears to be stylized or **handwritten**. However, the **blurriness** may affect readability,especially if viewed from a distance or in **smaller sizes**....

This picture shows the letters **"KFC"** made from **chicken nuggets**.......It's **very creative and funny, with a simple design**........which has a quality assessment score of **4**.....

Figure 6. Qualitative results of the font quality assessment.

the DFD dataset, more feedback data significantly reduce Font-Agent's hallucination rate and segments, proving that detailed human feedback effectively aligns MLLM behavior. Fig. 6 highlights the superior font understanding of Font-Agent compared to existing models. This work advances practical font recognition and quality assessment. However, Font-Agent performance can improve with ongoing benchmark efforts. We expect more feedback data to further boost performance and drive future research.

5. Conclusion

Hallucination continues to be a major obstacle to the practical implementation of MLLMs. To address this issue, we propose Font-Agent, an innovative framework specifically developed to enhance the reliability and interpretability of font-quality evaluation and recognition across multiple languages. By incorporating fine-grained human feedback, our model is closely aligned with user expectations, substantially reducing errors in long-form font evaluations while maintaining both accuracy and usefulness. Experimental results indicate that Font-Agent consistently surpasses existing models in terms of both reliability and cross-language recognition. Future research will focus on systematically expanding dataset coverage and developing robust validation protocols across a variety of practical application domains.

6. Acknowledgment

This work is supported by the National Natural Science Foundation of China (No. 62276221, No. 62376232), the Fujian Provincial Natural Science Foundation of China (No. 2022J01002).

References

[1] Darius Afchar, Vincent Nozick, Junichi Yamagishi, and Isao Echizen. Mesonet: a compact facial video forgery detection network. In *Proceedings of the IEEE International Workshop on Information Forensics and Security*, 2018. 2, 6

[2] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023. 4, 6

[3] Darwin Bautista and Rowel Atienza. Scene text recognition with permuted autoregressive sequence models. In *European Conference on Computer Vision*, pages 178–196. Springer, 2022. 8

[4] Zhanzhan Cheng, Fan Bai, Yunlu Xu, Gang Zheng, Shiliang Pu, and Shuigeng Zhou. Focusing attention: Towards accurate text recognition in natural images. In *Proceedings of the IEEE international conference on computer vision*, pages 5076–5084, 2017. 6

[5] Chee Kheng Ch'ng and Chee Seng Chan. Total-text: A comprehensive dataset for scene text detection and recognition. In *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*, pages 935–942. IEEE, 2017. 6

[6] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017. 1, 6

[7] Alexandre Ciancio, A. L. N. T. Targino da Costa, E. A. B. da Silva, Amir Said, Ramin Samadani, and Pere Obrador. No-reference blur assessment of digital pictures based on multifeature classifiers. *IEEE Transactions on Image Processing*, 20(1):64–75, 2011. 6

[8] Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Bin Wang, Linke Ouyang, Xilin Wei, Songyang Zhang, Haodong Duan, Maosong Cao, et al. InternLM-XComposer2: Mastering free-form text-image composition and comprehension in vision-language large model. *CoRR*, abs/2401.16420, 2024. 6, 7

[9] Shancheng Fang, Hongtao Xie, Yuxin Wang, Zhendong Mao, and Yongdong Zhang. Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7098–7107, 2021. 8

[10] Deepti Ghadiyaram and Alan C. Bovik. Massive online crowdsourced study of subjective and objective picture quality. *IEEE Transactions on Image Processing*, 25(1):372–387, 2016. 6

[11] Franz Götz-Hahn, Vlad Hosu, Hanhe Lin, and Dietmar Saupe. Konvid-150k: A dataset for no-reference video quality assessment of videos in-the-wild. In *IEEE Access 9*. IEEE, 2021. 6, 7

[12] Haibin He, Xinyuan Chen, Chaoyue Wang, Juhua Liu, Bo Du, Dacheng Tao, and Qiao Yu. Diff-font: Diffusion model for robust one-shot font generation. *International Journal of Computer Vision*, pages 1–15, 2024. 3

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 6

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 1

[15] David Holz. Midjourney. url = https://www.midjourney.com/, 2023. 1

[16] Vlad Hosu, Hanhe Lin, Tamas Sziranyi, and Dietmar Saupe. KonIQ-10k: An ecologically valid database for deep learning of blind image quality assessment. *IEEE Transactions on Image Processing*, 29:4041–4056, 2020. 6

[17] Heng Huang, Xin Jin, Yaqi Liu, Hao Lou, Chaoen Xiao, Shuai Cui, Xinning Li, and Dongqing Zou. Predicting scores of various aesthetic attribute sets by learning from overall score labels. *arXiv preprint arXiv:2312.03222*, 2023. 6, 7

[18] Huggingface. Introducing idefics: An open reproduction of state-of-the-art visual language model, 2023. 6, 7

[19] Qing Jiang, Jiapeng Wang, Dezhi Peng, Chongyu Liu, and Lianwen Jin. Revisiting scene text recognition: A data perspective. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20543–20554, 2023. 8

[20] Yue Jiang, Zhouhui Lian, Yingmin Tang, and Jianguo Xiao. Scfont: Structure-guided chinese font generation via deep stacked networks. In *Proceedings of the AAAI conference on artificial intelligence*, pages 4015–4022, 2019. 1

[21] Dimosthenis Karatzas, Faisal Shafait, Seiichi Uchida, Masakazu Iwamura, Lluis Gomez i Bigorda, Sergi Robles Mestre, Joan Mas, David Fernandez Mota, Jon Almazan Almazan, and Lluis Pere De Las Heras. Icdar 2013 robust reading competition. In *2013 12th international conference on document analysis and recognition*, pages 1484–1493. IEEE, 2013. 6

[22] Dimosthenis Karatzas, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman Ghosh, Andrew Bagdanov, Masakazu Iwamura, Jiri Matas, Lukas Neumann, Vijay Ramaseshan Chandrasekhar, Shijian Lu, et al. Icdar 2015 competition on robust reading. In *2015 13th international conference on document analysis and recognition (ICDAR)*, pages 1156–1160. IEEE, 2015. 6

[23] Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. MUSIQ: Multi-scale image quality transformer. In *IEEE International Conference on Computer Vision*, pages 5148–5157, 2021. 6, 7

[24] Junjie Ke, Keren Ye, Jiahui Yu, Yonghui Wu, Peyman Milanfar, and Feng Yang. Vila: Learning image aesthetics from user

comments with vision-language pretraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10041–10051, 2023. 6, 7

[25] Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:36652–36663, 2023. 2

[26] Eric C Larson and Damon M Chandler. Most apparent distortion: Full-reference image quality assessment and the role of strategy. *Journal of Electronic Imaging*, 19(1):011006–011006, 2010. 6

[27] Junyeop Lee, Sungrae Park, Jeonghun Baek, Seong Joon Oh, Seonghyeon Kim, and Hwalsuk Lee. On recognizing texts of arbitrary shapes with 2d self-attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 546–547, 2020. 8

[28] Hui Li, Peng Wang, Chunhua Shen, and Guyu Zhang. Show, attend and read: A simple and strong baseline for irregular text recognition. In *Proceedings of the AAAI conference on artificial intelligence*, pages 8610–8617, 2019. 8

[29] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning*, pages 19730–19742, 2023. 6

[30] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 6

[31] Hanhe Lin, Vlad Hosu, and Dietmar Saupe. KADID-10k: A large-scale artificially distorted IQA database. In *International Conference on Quality of Multimedia Experience*, pages 1–3, 2019. 6

[32] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *CoRR*, abs/2310.03744, 2023. 7

[33] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, pages 1–25, 2024. 6

[34] Ming-Yu Liu, Xun Huang, Arun Mallya, Tero Karras, Timo Aila, Jaakko Lehtinen, and Jan Kautz. Few-shot unsupervised image-to-image translation. In *Proc. ICCV*, pages 10551–10560, 2019. 2

[35] Yitian Liu and Zhouhui Lian. Qt-font: High-efficiency font synthesis via quadtree-based diffusion models. In *SIGGRAPH 2024 Conference Papers*, 2024. 1

[36] Yuliang Liu, Lianwen Jin, Shuaitao Zhang, Canjie Luo, and Sheng Zhang. Curved scene text detection via transverse and longitudinal sequence connection. *Pattern Recognition*, 90: 337–345, 2019. 6

[37] Yifang Men, Zhouhui Lian, Yingmin Tang, and Jianguo Xiao. Dyntypo: Example-based dynamic text effects transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5870–5879, 2019. 3

[38] Anand Mishra, Karteek Alahari, and CV Jawahar. Top-down and bottom-up cues for scene text recognition. In *2012 IEEE conference on computer vision and pattern recognition*, pages 2687–2694. IEEE, 2012. 6

[39] Kunato Nishina and Yusuke Matsui. Svgeditbench: A benchmark dataset for quantitative assessment of llm's svg editing capabilities. *arXiv preprint arXiv:2404.13710*, 2024. 3

[40] OpenAI. Gpt-4 technical report. *ArXiv*, abs/2303.08774, 2023. 6

[41] Song Park, Sanghyuk Chun, Junbum Cha, Bado Lee, and Hyunjung Shim. Few-shot font generation with localized style representations and factorization. In *Proc. AAAI*, pages 2393–2402, 2021. 2

[42] Song Park, Sanghyuk Chun, Junbum Cha, Bado Lee, and Hyunjung Shim. Multiple heads are better than one: Few-shot font generation with multiple localized experts. In *ICCV*, pages 13900–13909, 2021. 2

[43] Trung Quy Phan, Palaiahnakote Shivakumara, Shangxuan Tian, and Chew Lim Tan. Recognizing text with perspective distortion in natural scenes. In *Proceedings of the IEEE international conference on computer vision*, pages 569–576, 2013. 6

[44] Guo Pu, Shiyao Xu, Xixin Cao, and Zhouhui Lian. Dynamic texture transfer using patchmatch and transformers. *arXiv preprint arXiv:2402.00606*, 2024. 3

[45] Zeju Qiu, Weiyang Liu, Haiwen Feng, Zhen Liu, Tim Z Xiao, Katherine M Collins, Joshua B Tenenbaum, Adrian Weller, Michael J Black, and Bernhard Schölkopf. Can large language models understand symbolic graphics programs? *arXiv preprint arXiv:2408.08313*, 2024. 1, 3

[46] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763, 2021. 2

[47] Anhar Risnumawan, Palaiahankote Shivakumara, Chee Seng Chan, and Chew Lim Tan. A robust arbitrary text detection system for natural scene images. *Expert Systems with Applications*, 41(18):8027–8048, 2014. 6

[48] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF Conference on International Conference on Computer Vision*, 2019. 2, 6

[49] Hamid R. Sheikh, Muhammad F. Sabir, and Alan C. Bovik. A statistical evaluation of recent full reference image quality assessment algorithms. *IEEE Transactions on Image Processing*, 15(11):3440–3451, 2006. 6

[50] Fenfen Sheng, Zhineng Chen, and Bo Xu. Nrtr: A no-recurrence sequence-to-sequence model for scene text recognition. In *2019 International conference on document analysis and recognition (ICDAR)*, pages 781–786. IEEE, 2019. 8

[51] Baoguang Shi, Mingkun Yang, Xinggang Wang, Pengyuan Lyu, Cong Yao, and Xiang Bai. Aster: An attentional scene text recognizer with flexible rectification. *IEEE transactions on pattern analysis and machine intelligence*, 41(9):2035–2048, 2018. 8

[52] Cunzhao Shi, Chunheng Wang, Baihua Xiao, Song Gao, and Jinlong Hu. End-to-end scene text recognition using tree-structured models. *Pattern Recognition*, 47(9):2853–2866, 2014. 6

[53] Hossein Talebi and Peyman Milanfar. Nima: Neural image assessment. *IEEE TIP*, 2018. 6, 7

[54] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the International Conference on Machine Learning*, pages 6105–6114. PMLR, 2019. 1, 2, 6

[55] Licheng Tang, Yiyang Cai, Jiaming Liu, Zhibin Hong, Mingming Gong, Minhu Fan, Junyu Han, Jingtuo Liu, Errui Ding, and Jingdong Wang. Few-shot font generation by learning fine-grained local styles. In *Proc. CVPR*, pages 7895–7904, 2022. 2

[56] Maham Tanveer, Yizhi Wang, Ali Mahdavi-Amiri, and Hao Zhang. Ds-fusion: Artistic typography via discriminated and stylized diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 374–384, 2023. 3

[57] Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. Maxvit: Multi-axis vision transformer. *ECCV*, 2022. 6, 7

[58] Andreas Veit, Tomas Matera, Lukas Neumann, Jiri Matas, and Serge Belongie. Coco-text: Dataset and benchmark for text detection and recognition in natural images. *arXiv preprint arXiv:1601.07140*, 2016. 6

[59] Changshuo Wang, Lei Wu, Xiaole Liu, Xiang Li, Lei Meng, and Xiangxu Meng. Anything to glyph: Artistic font synthesis via text-to-image diffusion model. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–11, 2023. 3

[60] Chi Wang, Min Zhou, Tiezheng Ge, Yuning Jiang, Hujun Bao, and Weiwei Xu. Cf-font: Content fusion for few-shot font generation. In *Proc. CVPR*, pages 1858–1867, 2023. 2

[61] Jianyi Wang, Kelvin CK Chan, and Chen Change Loy. Exploring clip for assessing the look and feel of images. In *AAAI*, 2023. 6, 7

[62] Yizhi Wang and Zhouhui Lian. Deepvecfont: Synthesizing high-quality vector fonts via dual-modality learning. *ACM Transactions on Graphics (TOG)*, 40(6):1–15, 2021. 2

[63] Yuxin Wang, Hongtao Xie, Shancheng Fang, Jing Wang, Shenggao Zhu, and Yongdong Zhang. From two to one: A new scene text recognizer with visual language modeling network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14194–14203, 2021. 6

[64] Haoning Wu, Zicheng Zhang, Weixia Zhang, Chaofeng Chen, Chunyi Li, Liang Liao, Annan Wang, Erli Zhang, Wenxiu Sun, Qiong Yan, Xiongkuo Min, Guangtao Zhai, and Weisi Lin. Q-align: Teaching lmms for visual scoring via discrete text-defined levels. *arXiv preprint arXiv:2312.17090*, 2023. Equal Contribution by Wu, Haoning and Zhang, Zicheng. Corresponding Authors: Zhai, Guangtao and Lin, Weisi. 6, 7

[65] Haoning Wu, Hanwei Zhu, Zicheng Zhang, Erli Zhang, Chaofeng Chen, Liang Liao, Chunyi Li, Annan Wang, Wenxiu Sun, Qiong Yan, et al. Towards open-ended visual quality comparison. *CoRR*, abs/2402.16641, 2024. 6, 7

[66] Xudong Xie, Ling Fu, Zhifei Zhang, Zhaowen Wang, and Xiang Bai. Toward understanding wordart: Corner-guided transformer for scene text recognition. 2022. 3, 6

[67] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation, 2023. 2

[68] Xu Yan, Zhihao Yuan, Yuhao Du, Yinghong Liao, Yao Guo, Zhen Li, and Shuguang Cui. Clevr3d: Compositional language and elementary visual reasoning for question answering in 3d real-world scenes. *arXiv preprint arXiv:2112.11691*, 2021. 1

[69] Qiangpeng Yang, Jun Huang, and Wei Lin. Swaptext: Image based texts transfer in scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14700–14709, 2020. 3

[70] Zhenhua Yang, Dezhi Peng, Yuxin Kong, Yuyi Zhang, Cong Yao, and Lianwen Jin. Fontdiffuser: One-shot font generation via denoising diffusion with multi-scale content aggregation and style contrastive learning. *arXiv preprint arXiv:2312.12142*, 2023. 2, 3

[71] Li Sun Yangchen Xie, Xinyuan Chen and Yue lu. Dg-font: Deformable generative networks for unsupervised font generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021. 2, 3

[72] Mingshuai Yao, Yabo Zhang, Xianhui Lin, Xiaoming Li, and Wangmeng Zuo. Vq-font: Few-shot font generation with structure-aware enhancement and quantization. In *AAAI*, 2024. 3

[73] Q. Ye, H. Xu, G. Xu, J. Ye, M. Yan, Y. Zhou, J. Wang, A. Hu, P. Shi, Y. Shi, and *et al.* mPLUG-Owl: Modularization empowers large language models with multimodality. *CoRR*, abs/2304.14178, 2023. 6

[74] Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Haowei Liu, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. mPLUG-Owl2: Revolutionizing multi-modal large language model with modality collaboration. *CoRR*, abs/2311.04257, 2023. 7

[75] Zitong Yu, Chenxu Zhao, Zezheng Wang, Yunxiao Qin, Zhuo Su, Xiaobai Li, Feng Zhou, and Guoying Zhao. Searching central difference convolutional networks for face anti-spoofing. In *CVPR*, 2020. 5

[76] Lingjun Zhang, Xinyuan Chen, Yaohui Wang, Yue Lu, and Yu Qiao. Brush your text: Synthesize any scene text on images via diffusion model. 2023. 3

[77] Weixia Zhang, Guangtao Zhai, Ying Wei, Xiaokang Yang, and Kede Ma. Blind image quality assessment via vision-language correspondence: A multitask learning perspective. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 14071–14081, 2023. 6, 7

[78] Zhen Zhao, Jingqun Tang, Chunhui Lin, Binghong Wu, Can Huang, Hao Liu, Xin Tan, Zhizhong Zhang, and Yuan Xie. Multi-modal in-context learning makes an ego-evolving scene text recognizer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15567–15576, 2024. 5, 8