Advancing WordArt-Oriented Scene Text Recognition: Datasets and Methods

Xingsong Ye[1,2], Yongkun Du[1,2], Jiaxin Zhang[3], Haojie Zhang[4], Chong Sun[3], Chen Li[3], Jing Lyu[3], and Zhineng Chen[1,2(✉)]

[1] Institute of Trustworthy Embodied AI, Fudan University, China
[2] Shanghai Key Laboratory of Multimodal Embodied AI, Fudan University, China
[3] WeChat Vision, Tencent Inc., China
[4] South China University of Technology, China

Abstract. WordArt (artistic text) features highly customized fonts, textures, and layouts, making WordArt-oriented scene TExt Recognition (WATER) substantially more challenging than general Scene Text Recognition (STR). Existing STR datasets and methods, typically built around regular scene text and fixed-template inputs, struggle to scale to WATER. Thus, we aim to advance this task from both data and model perspectives. On the data side, we construct a 2M synthetic dataset, **WATER-S**, with the scale improved by hundreds of times compared to existing artistic text data. WATER-S consists of two complementary subsets. One rendered by an upgraded rendering pipeline (SynthWordArt), which provides highly accurate and controllable synthetic WordArt data. The other is generated by combining Qwen3-VL for prompt mining and Z-Image for image synthesis, which improves the coverage of realistic and diverse data. On the model side, we propose **WATERec**. It adopts a visual encoder supporting arbitrary-shaped inputs and an autoregressive decoder to model complex layouts, structurally breaking the bottleneck of fixed-template STR on WordArt. Experiments show that this architecture outperforms prior STR methods, achieving state-of-the-art performance on irregular texts such as WordArt. Together with **WATER-R**, carefully reorganized from existing real STR data, our strong baseline with the new synthetic data and model design reaches 90.40% accuracy on WordArt-Bench, surpassing both general-purpose and OCR-specialized vision-language models by a large margin. Code and data are available at https://github.com/YesianRohn/WATER.

Keywords: scene text recognition · data synthesis · artistic text

1 Introduction

WordArt refers to artistic text [49] that exhibits strong visual stylization and appears widely in posters, signboards, magazines, and other advertising design scenarios. Compared to regular scene text, artistic text is highly customized in

✉ Corresponding author. E-mail: zhinchen@fudan.edu.cn

Fig. 1: Top: Two subsets' examples of our synthetic datasets (WATER-S) and the real artistic text benchmark (WordArt-Bench). **Bottom:** Recognition accuracy on WordArt-Bench for various STR methods and VLMs. For each STR entry, the term before "/" denotes the model and the term after "/" denotes the training data used. "R" indicates existing real datasets only, while "RS" indicates adding our synthetic data.

font shape, texture, layout, and pattern fusion, as shown in the top of Fig. 1. Designers often embed complex graphics and semantic elements inside character contours, so that the text serves as both a linguistic carrier and a visual symbol. This highly stylized design introduces substantial visual distractions unrelated to the underlying text, making WordArt-oriented scene TExt Recognition (WATER) more challenging than general Scene Text Recognition (STR). Consequently, WATER is often treated as a distinct subtask within STR, and existing work [25, 49] typically includes dedicated test subsets and benchmarks for this setting. Although mainstream STR models [13, 15, 17, 60] and recent general/OCR-specialized vision-language models (VLMs) [1, 5, 40] perform well on common STR benchmarks, their performance on artistic text remains far from satisfactory (see the bottom of Fig. 1).

The primary bottleneck for WATER is the lack of data. Real artistic text is difficult to collect in real scenes, and its annotation is expensive and inconsistent.

Annotators must carefully inspect characters under complex textures and often must reason about ambiguous shapes to determine the correct label. As a result, current training sets are (i) too small to optimize modern models (e.g., WordArt provides only 4,805 training images [49]), and (ii) insufficiently diverse to cover the long-tail of styles seen in real designs. Breaking this bottleneck requires moving beyond purely manual collection, and large-scale synthesis becomes an almost inevitable choice.

However, current methods for scene text data generation [21,24,54] mainly focus on regular text and do not suit WordArt. Synthesizing artistic text is challenging for it must cover broad stylistic variations (fonts, textures, shadow) and realistic effects while still ensuring accurate labels. Therefore, we systematically explore two synthetic data paradigms oriented to WordArt: an improved tool-based rendering approach and a pioneering generative-model-based synthesis pipeline. They emphasize different aspects and, when combined, provide a richer and more comprehensive synthetic dataset **WATER-S**. Here, we customize and evaluate both paradigms for artistic text. (1) We collect about 11K open-source artistic fonts, and then develop an artistic-text-oriented rendering engine, **SynthWordArt**. It pastes target text with specified fonts and layout rules onto background images. Based on this tool, we generate a 1M-scale synthetic artistic text dataset, **WATER-T**, which substantially enlarges the number of training samples and the diversity of fonts. (2) To better mimic the rich style semantics and content compositions in real design scenarios, we leverage the advanced VLM, Qwen3-VL-8B [1], to automatically mine and generate fine-grained captions from existing artistic text images, and obtain about 273K high-quality textual prompts. Then, we employ the state-of-the-art (SOTA) open-source image generation model Z-Image-Turbo [3] to synthesize images containing artistic-style text based on these prompts. Finally, we construct another 1M-scale synthetic artistic text dataset, **WATER-Z**. Compared with traditional rendering, WATER-Z exhibits higher diversity and realism in background textures, layout composition, and global visual style, as depicted at the top of Fig. 1.

Beyond the data, we aim to further establish a strong baseline specifically for WordArt. We find that WordArt often contains highly irregular text with unfixed aspect ratios. But many STR models resize all inputs to a fixed shape, which can severely distort highly irregular text. SVTRv2 [13] shows that such resizing can dominate failure cases and proposes multi-shape resizing with predefined templates. However, this approach requires manual template design and lacks flexibility in WordArt. Motivated by modern VLMs, we introduce a NaViT-like encoder [7] with RoPE [22,39] to support arbitrary-shaped inputs without relying on predefined resizing templates. On the decoding side, WordArt often violates standard left-to-right reading order and requires stronger contextual reasoning. Thus we adopt an autoregressive (AR) decoder to generate the character sequence step by step. Combining these ideas, we present **WATERec**, a strong STR baseline that better fits the characteristics of the WordArt data.

To fairly and comprehensively evaluate STR methods on artistic text, we also re-construct **WATER-R** from existing real scene text data (Union14M-

L [25], WordArt-Train [49], WAS-R [50]). We perform strict hashing deduplication, so as to avoid label leakage between training and evaluation. Experimental results show that WATERec achieves new SOTA performance on the WordArt-Bench [49] and on several highly challenging subsets (including artistic subset) of the Union14M-Benchmark [25]. When we augment training with our synthetic data, adding either WATER-T or WATER-Z alone already brings consistent and significant improvements. When combining both datasets during training, WATERec further improves its accuracy on the WordArt-Bench to 90.40%, which, to our knowledge, is the first result exceeding 90% on this benchmark. We also evaluate representative general VLMs [1,8,42] and OCR-specialized VLMs [5,40,43–45] on the WordArt-Bench. Their best accuracies are 78.36% and 81.54%, respectively, which remain notably lower than that of our expert baseline. This gap highlights the necessity of dedicated architectural and data design for artistic text.

In summary, our main contributions are:

- We build a large-scale synthetic artistic text dataset, WATER-S, consisting of the tool-rendered subset and the model-generated subset. This dataset provides complementary diversity in font style, layout, and visual texture, and offers a scalable data foundation for future research.
- We design WATERec to support arbitrary-shaped inputs in STR for the first time and to better model artistic text with complex layouts.
- We systematically evaluate representative methods on various STR benchmarks. Combined with WATER-R, WATER-S and WATERec, our strong baseline achieves new SOTA results on WordArt and other challenging scenes.

2 Related Work

2.1 Model Perspective on WATER

From a more general STR perspective, mainstream model architectures can be roughly divided by decoding strategy into Connectionist Temporal Classification [18] (CTC)-based, Parallel Decoding (PD)-based, and AR-based models. CTC-based methods [12, 13, 38, 58] convert 2D images into 1D character sequences via CTC alignment, and work well for regular, horizontally aligned text. Recent variants like SVTRv2 [13] improve robustness to irregular text by using multi-scale resizing and CTC feature rearrangement. PD-based methods [11, 16, 34, 46, 53, 57, 59] can be viewed as NAR models built on attention mechanisms, predicting all characters in parallel, offering high efficiency but struggling with complex artistic layouts where reading order is ambiguous. AR-based methods [2,10,19,20,29,30,35,37] decode characters step by step (typically left to right), better handling artistic or heavily distorted text through iterative context refinement. CornerTransformer [49] adopts this AR paradigm and is the first baseline tailored to artistic text recognition. On the general-purpose encoder side, vision backbones have progressed from ViT [9] to NaViT [7] to support arbitrary aspect ratios, often combined with RoPE [22, 39] and widely used in

modern VLMs [1,5,40]. Inspired by this, we use it to support arbitrary-shaped input, paired with AR decoding as our baseline.

2.2 Data Perspective on WATER

Early STR datasets (IIIT5K [32], ICDAR13 [27], SVT [41]) focus on regular text in simple conditions. Later benchmarks add irregular text: blurred (ICDAR15 [26]), perspective-distorted (SVTP [33]), and curved (CUTE80 [36]) to better match real-world cases. WordArt [49] is the first dataset and competition [48] dedicated to artistic text images, derived from TextSeg [52]. WAS-R [50] extends this with more real artistic text. Union14M [25] revisits STR from a data perspective: it collects a large real-world training set and builds a challenging benchmark of seven scenarios, with artistic text as one subset. Besides real images, synthetic data [14,21,24,55,56] plays a crucial role in STR. Earlier methods use tool engines to render mainly regular text, without focusing on artistic styles. Newer work [54,61] uses diffusion models to generate more diverse and realistic STR data. Here, we follow both directions, combining traditional rendering and diffusion-based generation to build our synthetic WordArt data.

3 Synthetic Dataset: WATER-S

Due to the scarcity of real-world artistic text data, training models solely on existing real datasets already reaches a performance bottleneck. However, collecting additional large-scale, high-quality artistic text in the wild is costly and impractical. This limitation makes synthetic data a key component for scaling up training. We therefore design two synthetic data suites that explicitly target two core objectives: controllability and diversity. Along two complementary paths, (1) tool-based rendering and (2) model-based generating, we construct two large-scale synthetic datasets, **WATER-T** and **WATER-Z**, collectively referred to as **WATER-S**. In this section, we describe the construction pipeline, design choices, and statistical properties of WATER-S.

3.1 Tool-based Rendering: WATER-T

Building on SynthText [21] and SynthTIGER [56], we design **SynthWordArt**, a rendering engine tailored to artistic text. Our goal is to preserve the precise controllability of classical rendering pipelines over text content and layout, while more faithfully mimicking the diverse layouts and visual distractions that appear in real design scenarios.

Compared with the original synthesis tools, SynthWordArt introduces the following key modifications: (1) Artistic font support. We replace the commonly used Google Fonts with our library of 11,250 artistic fonts, covering a wide spectrum of artistic styles. (2) Enhanced layout and typography. Beyond the standard horizontal layout, we add several layout patterns that frequently appear

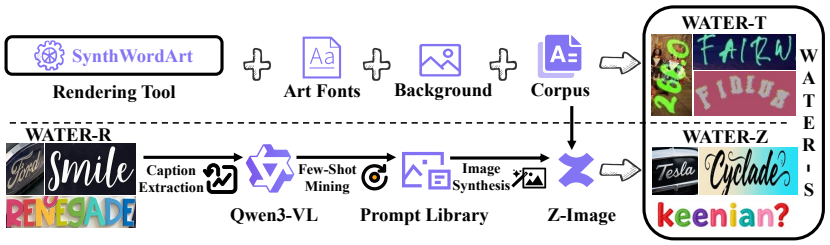

Fig. 2: Pipelines of our two synthetic datasets. The top illustrates the WATER-T synthesis pipeline: the art-text-oriented tool SynthWordArt renders provided artistic fonts, background images, and real text corpora into artistic text images. The bottom shows the WATER-Z synthesis pipeline: A small set of real artistic text images are fed into Qwen3-VL (8B) [1] to obtain detailed captions. Then Qwen3-VL is leveraged in a few-shot manner to expand these captions into a Prompt Library. Finally Z-Image (Turbo) [3] use these prompts to generate more artistic text images.

in artistic design, including curved lines, vertical text, multi-orientation layouts, and geometric transformations such as perspective and stretching.

The rest of the pipeline (as shown in the top of Fig. 2) follows standard rendering tools. For each sample, we randomly sample a text string, a font, a layout configuration, and a background image, and paste the rendered text onto the background. The layout configuration is drawn with explicit sampling ratios, approximately 0.2 for curved text, 0.3 for multi-oriented text, and the remainder for general (near-horizontal) layouts. Randomization over these factors yields rich diversity in visual appearance and layout. Using SynthWordArt, we generate 1M synthetic images, denoted as **WATER-T**.

3.2 Model-based Generating: WATER-Z

While tool-based rendering offers strong controllability and perfect text correctness, it still lags behind real designs in terms of global stylistic coherence, semantic texture integration, and overall visual naturalness. To further enhance diversity and realism, we attempt to use the mainstream generative model to synthesize data.

The effectiveness of image generation heavily depends on prompt quality. Simple, template-based prompts are insufficient to cover the rich visual semantics of real design. We therefore propose an automatic few-shot mining pipeline (see Fig. 2) for artistic text generation. Starting from 31,335 samples in the WordArt [49] and WAS-R [50] training set, we use a VLM (e.g., Qwen3-VL-8B [1]) to analyze each artistic text image and generate a detailed caption (please refer to the appendix for the prompt and caption examples). The caption additionally uses an editable placeholder (e.g., `<Text>`) for the specific word content. Given these captions, we randomly sample few-shot (e.g., 3) examples and ask the VLM to imitate them and produce a new prompt. We then apply strict filtering and deduplication to ensure that each prompt contains an editable placeholder

and to remove near-duplicate prompts. This pipeline yields 273,488 high-quality prompts tailored for artistic text generation. The prompt resource is independent of any particular generation model, so more advanced models (such as Nano-Banana2) can leverage it to produce even better results.

Here, we adopt Z-Image-Turbo [3], an open-source generative model, as our backbone generator. We use the official default configuration and set the output resolution to 256×256 for balancing visual fidelity and generation efficiency. Using the prompts described above and randomly substituting the text placeholder with target strings, we synthesize 1M images to construct **WATER-Z**.

3.3 Analysis of WATER-T and WATER-Z

Although WATER-T and WATER-Z arise from very different generation mechanisms, they complement each other along several important dimensions: (1) Font control and text accuracy. WATER-T renders text directly with explicit artistic fonts, so the character shapes and text content are accurate and exactly controllable. In contrast, WATER-Z excels at generating complex materials and globally coherent styles, but its character shapes and legibility are less predictable. (2) Layout controllability and semantic naturalness. WATER-T allows explicit control over layout patterns and geometric transformations, which enables systematic analysis of how different layouts affect recognition performance. WATER-Z, on the other hand, produces layouts that more closely resemble real designer creations, with more natural composition and stronger semantic integration between foreground text and background. (3) Distribution coverage. Jointly using WATER-T and WATER-Z for training covers both the "strongly controlled" and "style-diverse" regimes. This broad coverage is crucial for improving generalization to real artistic text in the wild. It is worth noting that, although the current versions of both datasets focus on English WordArt, the entire generation pipeline is language-agnostic and can be directly adapted to multilingual scenarios by replacing the underlying text corpus.

4 Strong Baseline: WATERec

4.1 Overall Architecture

WATERec aims to provide a simple yet effective baseline for artistic text recognition. The model (as shown in Fig. 3) is fully generic: it does not rely on any fixed-shape template, it handles images with arbitrary aspect ratios, and it models highly unconstrained layouts and visual styles in a unified way.

As illustrated in Fig. 3, WATERec adopts a simple encoder design that accepts inputs of arbitrary shape, followed by an AR Transformer decoder. The encoder is a 6-layer Transformer with RoPE attention [22,39] that processes visual token sequences of varying length (see details below), and the decoder uses 2 cross-attention layers to attend to the encoder outputs and predicts characters one by one under a standard cross-entropy loss. This design improves robustness to non-standard reading orders, curved or circular layouts, vertical text, and other complex artistic typesetting styles.

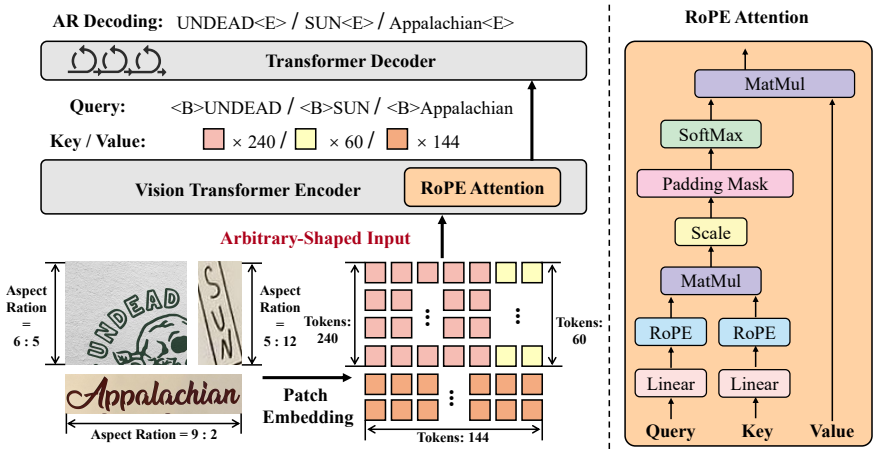

Fig. 3: The overall architecture of WATERec. It consists of a Vision Transformer Encoder that uses RoPE Attention (illustrated on the right) to support inputs of arbitrary shape, and an AR Transformer Decoder. It also shows images of different sizes, which are processed into tokens of different lengths. `<B>` denotes the beginning token of decoding, and `<E>` denotes the end token of decoding.

4.2 Arbitrary-Shaped Input

Most traditional STR methods [17, 25, 38] assume fixed-template inputs (e.g., 32×128), or rely on some predefined templates [13]. These designs work well for general horizontal text in natural scenes, but they show clear limitations in artistic text scenarios. Artistic text exhibits extreme variation in aspect ratio, ranging from very long horizontal titles to almost square or vertical text.

Therefore, WATERec adopts a visual encoding strategy that naturally adapts to arbitrary aspect ratios. Concretely, given an input image $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$ of arbitrary resolution, we first rescale it while preserving the aspect ratio to $\hat{\mathbf{I}} \in \mathbb{R}^{\hat{H} \times \hat{W} \times 3}$. The resulting number of tokens $N = (\hat{H}/p) \times (\hat{W}/p)$ lies in a predefined range $[64, 256]$, where $p = 4$ is the patch size. The upper bound ensures a fair comparison with existing methods and the lower bound prevents very small images from being represented with too few tokens. We then linearly project each image patch into a d-dimensional visual token ($d = 384$ to align with prior work), yielding a token sequence $\mathbf{X} \in \mathbb{R}^{N \times d}$. We arrange these patch tokens in a row-major order to obtain a 1D sequence. The WATERec encoder maps $\mathbf{X}$ to contextual features $\mathbf{Z} \in \mathbb{R}^{N \times d}$, which are then consumed by the AR decoder through cross-attention to autoregressively produce the character sequence $\{y_1, \ldots, y_T\}$ ($T \leq 25$). To support efficient batch training and robust inference, we apply padding to sequences of different lengths at the batch level and use the corresponding padding masks in attention computation.

A remaining challenge is how the encoder above should perceive the spatial layout of arbitrary-shaped inputs, which is crucial for recognizing text under

complex artistic typesetting. Absolute and sinusoidal position embeddings (APE and SPE) are highly sensitive to image size and shape, and training data cannot cover all resolutions and layouts. This motivates our choice of the RoPE design [22] for the visual encoder of WATERec. The core idea is to apply position-dependent rotations to the query (Q) and key (K) vectors, so that the dot-product attention implicitly encodes relative positional information and generalizes naturally across the variable token sequence length N. For more details of the three position embeddings, please refer to the appendix.

5 Experiments

5.1 Datasets and Implementation Details

Training data. In addition to the synthetic datasets introduced earlier, we construct a large-scale real-world training set by combining three sources: the widely used Union14M-L [25] for STR, the training split of WordArt [49], and the WAS-R [50]. To avoid label leakage caused by overlapping images between training and evaluation sets, we perform strict hashing deduplication between the merged real dataset and all test sets used in our experiments. The resulting real training set is denoted as **WATER-R** and contains 3,225,130 text instances.

Test data. For artistic text evaluation, we use the test split of WordArt [49] with 1,511 images and refer to this benchmark as **A-Bench**. To assess the generalization of WATERec, we also evaluate on six common benchmarks (collectively denoted as **C-Bench**): ICDAR 2013 (IC13) [27], Street View Text (SVT) [41], IIIT5K-Words (IIIT5K) [32], ICDAR 2015 (IC15) [26], Street View Text-Perspective (SVTP) [33], and CUTE80 (CUTE) [36]. In addition, we report results on the Union14M-Benchmark (**U-Bench**) [25], with seven subsets: Curve (*CUR*), Multi-Oriented (*MLO*), **Artistic** (*ART*), Contextless (*CTL*), Salient (*SAL*), Multi-Words (*MLW*), and General (*GEN*).

Training setup. We train WATERec and all comparison models (with their official configurations unchanged) using the OpenOCR framework. For WATERec, we largely follow the training protocol of SVTRv2 [13]. We use the AdamW optimizer [31] with a weight decay of 0.05, a base learning rate of 6.5×10^{-4}, and a total batch size of 2048. We adopt a one-cycle learning rate scheduler with 1.5 epochs of linear warm-up over 20 training epochs. Data augmentation follows PARSeq [2] and includes random rotation, perspective distortion, motion blur, and Gaussian noise. The maximum text length during training is 25. The character set contains 94 tokens, including digits, letters, and common symbols. All models are trained on 8 NVIDIA V100 GPUs.

5.2 Experimental Results

STR model comparison. Tab. 1 reports a systematic comparison of the three mainstream STR paradigms. Among NAR (CTC and PD) methods, SVTRv2 [13]

Type	Model	WordArt Benchmark	Common Benchmarks							Union14M-Benchmark							
			IIIT	*SVT*	*IC13*	*IC15*	*SVTP*	*CUTE*	*AVG*	*CUR*	*MLO*	*ART*	*CTL*	*SAL*	*MLW*	*GEN*	*AVG*
CTC	CRNN [38]	62.54	93.07	87.02	92.30	80.95	80.47	82.64	86.07	27.21	35.65	39.00	53.15	23.69	48.18	64.53	41.63
	SVTR [12]	76.96	97.30	94.74	96.38	87.47	89.30	93.75	93.16	68.38	75.46	62.67	71.50	66.20	71.60	77.00	70.40
	SVTRv2 [13]	86.56	99.17	**98.45**	**99.18**	90.23	94.73	**98.96**	96.79	90.48	89.26	79.78	**86.14**	85.66	86.29	85.37	86.14
PD	ABINet [17]	84.64	98.73	97.37	97.67	90.01	94.73	98.26	96.13	83.76	90.14	72.00	77.54	80.29	78.64	81.71	80.58
	LPV [39]	80.34	97.93	95.05	96.62	87.69	91.47	96.18	94.16	76.17	84.59	64.44	73.94	74.42	72.69	79.37	75.09
	BUSNet [46]	80.67	98.00	95.52	97.55	88.79	93.33	95.83	94.84	79.97	87.14	66.22	74.20	76.18	68.81	81.28	76.26
	CPPD [11]	81.60	97.00	96.29	97.43	87.02	92.71	96.18	94.44	85.20	84.30	72.00	77.92	79.66	75.12	80.99	79.31
AR	PARSeq [2]	84.51	98.40	97.68	98.13	89.40	95.50	97.57	96.11	84.71	92.26	74.67	80.23	80.35	79.61	84.02	82.26
	MAERec [25]	86.23	99.13	97.99	99.07	91.05	**96.28**	98.26	**96.96**	90.60	93.64	78.00	84.34	86.86	86.53	**85.65**	86.52
	OTE [51]	83.98	97.87	96.75	97.20	89.18	93.33	96.53	95.14	81.16	88.09	72.78	73.81	76.69	62.50	81.38	76.63
	SVTRv2-AR [13]	87.36	**99.23**	97.06	98.72	90.61	95.35	97.92	96.48	91.59	**95.25**	80.33	86.01	87.49	87.14	85.58	87.63
	WATERec	**88.55**	99.17	97.68	98.72	**91.11**	95.19	98.26	96.69	**93.03**	94.89	**80.56**	85.62	**89.39**	**88.35**	85.16	**88.14**

Table 1: Quantitative comparison of different STR models trained on the WATER-R dataset. The values are all percentages (%), and AVG represents the arithmetic mean.

achieves strong performance on both A-Bench and U-Bench, benefiting from its design for handling multi-scale inputs. Building on this architecture, its AR variant SVTRv2-AR further improves accuracy on artistic text, indicating that coupling attention-based sequence modeling with strong visual features is especially beneficial in complex scenes. WATERec follows the AR paradigm but preserves the original aspect ratio of artistic text images as much as possible. This design yields the best average accuracy on both A-Bench and U-Bench, demonstrating that retaining the native width/height ratio is particularly effective for challenging subsets such as Artistic and Salient. However, this comes with a trade-off: on the more regular and simple C-Bench, the fixed-resolution MAERec [25] slightly outperforms SVTRv2 and WATERec. This suggests that enforcing a unified input resolution remains advantageous in standard scenarios.

VLM capabilities. On A-Bench, we further evaluate mainstream OCR systems, including general VLMs (Qwen3-VL-8B [1], InternVL3.5-8B [42], Nemotron-VL-8B [8]) and OCR-specialized VLMs / general OCR tools (GOT-OCR 2.0 [43], PaddleOCR-VL [5], HunyuanOCR [40], DeepSeek-OCR [44], DeepSeek-OCR2 [45], PP-OCRv5 [6]). As shown in Fig. 1, their accuracies on A-Bench mostly fall in the 70%–80% range, with HunyuanOCR performing best at 81.54%. While VLMs already perform strongly in general OCR settings, they still lack robustness and fine-grained perception in artistic text scenarios with severe deformations, diverse styles, and complex layouts. This gap highlights artistic text recognition as a challenging and important direction that merits further study.

Fine-tuning VLMs with WATER-S. A natural question is whether a strong general VLM can close the gap on WordArt after task adaptation. To examine this, we fine-tune Qwen3-VL-8B [1] with LoRA [23] for 20k steps using the ms-swift framework and our data. As shown in Tab. 2, our data brings clear gains, confirming that our synthetic data is also useful for VLM adaptation. Nevertheless, the fine-tuned 8B VLM still does not surpass our lightweight expert

Model	#Params	Training Data	A-Bench
Qwen3-VL-8B (zero-shot)	8B	–	72.01
Qwen3-VL-8B + SFT	8B	WATER-R	82.59
Qwen3-VL-8B + SFT	8B	WATER-R+WATER-S	84.78
WATERec (ours)	∼26M	WATER-R+WATER-S	**90.40**

Table 2: Expert baseline vs. SFT on a strong general VLM, evaluated on A-Bench.

WATERec. This indicates that, a compact task-specialized recognizer remains more effective and efficient for difficult WordArt recognition.

Training Data	Volume	WordArt Benchmark	Common Benchmarks							Union14M-Benchmark							
			IIIT	*SVT*	*IC13*	*IC15*	*SVTP*	*CUTE*	*AVG*	*CUR*	*MLO*	*ART*	*CTL*	*SAL*	*MLW*	*GEN*	*AVG*
WATER-R	3.2M	88.55	99.17	97.68	98.72	91.11	95.19	98.26	96.69	93.03	94.89	80.56	85.62	89.39	88.35	85.16	88.14
+ WATER-T	+0.5M	89.54	99.33	98.30	98.83	90.72	95.35	98.61	96.86	93.98	95.69	82.44	86.26	**90.08**	89.44	85.52	89.06
+ WATER-T	+1M	89.81	99.33	**98.61**	**98.95**	90.89	96.43	98.96	**97.20**	94.27	96.20	83.22	87.16	90.02	89.44	85.60	89.42
+ WATER-Z	+0.5M	88.91	99.07	98.30	**98.95**	90.56	95.66	97.92	96.74	93.65	95.11	81.56	86.39	89.70	89.20	85.25	88.69
+ WATER-Z	+1M	89.41	99.17	**98.61**	98.37	90.89	96.12	97.57	96.79	93.57	94.96	80.56	84.98	89.51	89.56	85.06	88.31
+ WATER-S	+1M	89.94	99.33	98.45	**98.95**	**91.44**	95.66	98.61	97.07	94.27	95.25	83.11	86.65	89.58	89.56	85.40	89.12
+ WATER-S	+2M	**90.40**	**99.43**	98.30	98.48	91.22	95.66	98.96	97.01	94.27	95.54	**84.11**	86.91	89.89	89.32	85.60	89.38
+ WATER-S	+3M	90.07	99.17	98.15	98.83	90.83	96.12	98.96	97.01	**94.31**	**96.35**	83.11	**87.55**	89.83	89.32	**85.62**	**89.44**
+SynthText [21]	+1M	88.95	99.10	**98.61**	98.83	91.00	**96.59**	98.61	97.12	93.94	95.76	82.56	86.39	89.58	**89.81**	85.46	89.07
+ SynthTIGER [56]	+1M	88.55	99.23	98.30	98.83	91.28	95.35	**99.65**	97.11	93.49	94.89	80.33	86.26	88.76	89.08	85.47	88.32
+ TextSSR [54]	+1M	88.09	99.30	97.99	98.48	90.89	96.28	98.61	96.93	93.57	95.62	82.22	85.88	89.07	88.71	85.28	88.62
WATER-S	2M	80.94	97.53	93.97	96.97	80.23	86.67	95.14	91.75	77.29	81.52	71.22	81.90	69.80	83.25	54.76	74.25

Table 3: Performance comparison of the gains brought by synthetic data from different sources and at different scales to the real data.

Synthetic data comparison. Tab. 3 analyzes the effect of adding our synthetic artistic data on top of the real dataset WATER-R. We observe consistent and significant gains on both artistic and general benchmarks. Among different synthetic sources, WATER-S, which combines tool-rendered and model-based generation, shows the most robust performance across scales. On A-Bench, adding 2M WATER-S samples to WATER-R improves accuracy by 1.85%. On C-Bench and U-Bench, it also yields additional gains. These results indicate that moderately scaled, high-quality, and diverse artistic synthetic data is crucial for improving model performance. Comparing different synthetic types, WATER-T consistently outperforms conventional tool-rendered datasets such as SynthText [21] and SynthTIGER [56] on both artistic and general benchmarks. This shows that explicitly injecting artistic fonts and complex layouts into the traditional synthesis pipeline is both effective and necessary. At the same time, WATER-Z outperforms general diffusion-based STR data synthesis methods like TextSSR [54] on artistic text, demonstrating the necessity of WordArt data generation. Moreover, WATER-T and WATER-Z are naturally complementary in visual style and scene diversity. For example, 1M WATER-S constructed from 0.5M WATER-T and

0.5M WATER-Z outperforms each of their respective 1M counterparts. Training on WATER-S alone still delivers strong performance, further demonstrating the potential of our synthetic data.

5.3 Ablation Study

Scale of synthetic data. We first study the marginal gains of WATER-S at different scales, as reported in Tab. 3. The improvement from synthetic data follows a "rise-then-saturate" pattern. When increasing WATER-S from 1M to 2M samples, the gains are consistent across A-Bench and most U-Bench subsets. It suggests that at this scale, the additional diversity in style, fonts, and scenes compensates well for the coverage gaps in real data while keeping noise under control. However, when further scaling WATER-S to 3M, the overall gains become marginal and some subsets even show slight fluctuations. Given that the real data size is fixed, too many synthetic samples introduce distribution mismatch and noise accumulation, which offsets the benefits. In our setting, a synthetic-to-real ratio of roughly 2:3 strikes a favorable balance between improving generalization and controlling noise.

Model	Training Data	WordArt	Common Benchmarks							Union14M-Benchmark							
		Benchmark	*IIIT*	*SVT*	*IC13*	*IC15*	*SVTP*	*CUTE*	*AVG*	*CUR*	*MLO*	*ART*	*CTL*	*SAL*	*MLW*	*GEN*	*AVG*
SVTRv2 [13]	WATER-R	86.56	99.17	98.45	99.18	90.23	94.73	98.96	96.79	90.48	89.26	79.78	86.14	85.66	86.29	85.37	86.14
	+WATER-S	88.68	99.33	98.15	98.83	90.89	95.19	99.31	96.95	92.50	92.62	82.11	87.68	88.63	89.08	84.86	88.21
	Δ	+2.12	+0.17	-0.31	-0.35	+0.66	+0.47	+0.35	+0.16	+2.02	+3.36	+2.33	+1.54	+2.97	+2.79	-0.51	+2.07
ABINet [17]	WATER-R	84.64	98.73	97.37	97.67	90.01	94.73	98.26	96.13	83.76	90.14	72.00	77.54	80.29	78.64	81.71	80.58
	+WATER-S	87.03	98.40	97.68	98.25	90.28	94.42	97.92	96.16	86.07	91.16	77.11	80.49	83.64	83.37	81.58	83.35
	Δ	+2.39	-0.33	+0.31	+0.58	+0.28	-0.31	-0.35	+0.03	+2.31	+1.02	+5.11	+2.95	+3.35	+4.73	-0.13	+2.76
SVTRv2-AR	WATER-R	87.36	99.23	97.06	98.72	90.61	95.35	97.92	96.48	91.59	95.25	80.33	86.01	87.49	87.14	85.58	87.63
	+WATER-S	90.14	99.17	98.30	99.18	91.28	95.50	98.61	97.01	94.19	95.69	84.67	86.26	89.51	89.56	85.65	89.36
	Δ	+2.78	-0.07	+1.24	+0.47	+0.66	+0.16	+0.69	+0.52	+2.60	+0.44	+4.33	+0.25	+2.02	+2.43	+0.07	+1.73
WATERec	WATER-R	88.55	99.17	97.68	98.72	91.11	95.19	98.26	96.69	93.03	94.89	80.56	85.62	89.39	88.35	85.16	88.14
	+WATER-S	90.40	99.43	98.30	98.48	91.22	95.66	98.96	97.01	94.27	95.54	84.11	86.91	89.89	89.32	85.60	89.38
	Δ	+1.85	+0.27	+0.62	-0.23	+0.11	+0.47	+0.69	+0.32	+1.24	+0.66	+3.56	+1.28	+0.51	+0.97	+0.43	+1.24

Table 4: Effect of adding our WATER-S (2M) synthetic data to different STR models.

Generality of synthetic data. We then validate the generality and robustness of WATER-S across different STR architectures, as shown in Tab. 4. We consider a CTC-based model (SVTRv2 [13]), a PD-based model (ABINet [17]), another AR-based model (SVTRv2-AR), and our WATERec. For all four models, adding WATER-S on top of the same real dataset WATER-R leads to consistent and significant improvements on A-Bench, with gains ranging from +1.85% to +2.78%. This cross-architecture benefit indicates that high-quality artistic synthetic data is broadly useful rather than tailored to a specific model design. On the more challenging U-Bench, all models also obtain 1%–4% improvements on most subsets, with especially notable gains on the Artistic subset.

Arbitrary Shape	Position Encoding	Token Range	WordArt Benchmark	Common Benchmarks			Union14M-Benchmark							
				Regular	Irregular	Average	*CUR*	*MLO*	*ART*	*CTL*	*SAL*	*MLW*	*GEN*	*AVG*
✗	NoPE	[256, 256]	86.83	98.25	95.15	96.70	90.15	94.52	77.78	84.34	86.73	87.86	85.41	86.69
✓	NoPE	[64, 256]	49.57	89.14	86.07	87.60	60.76	61.29	47.56	54.17	52.37	47.21	74.94	56.90
✓	APE	[64, 256]	87.69	98.29	94.41	96.35	92.00	93.86	79.00	85.24	88.63	86.89	84.08	87.10
✓	SPE	[64, 256]	87.29	98.36	94.37	96.37	92.25	94.08	79.00	84.21	88.12	86.17	84.29	86.87
✗	RoPE	[256, 256]	87.88	**98.71**	94.99	96.85	92.09	90.94	80.22	85.11	87.87	87.14	**85.55**	86.99
✓	RoPE	[1, 256]	88.29	98.47	94.80	96.64	93.36	94.96	80.11	**85.62**	**89.58**	89.20	78.81	87.38
✓	RoPE	[64, 256]	88.55	98.52	94.86	96.69	93.03	94.89	80.56	**85.62**	89.39	88.35	85.16	88.14
✓	RoPE	[64, 512]	**88.82**	98.61	**95.75**	**97.18**	**94.39**	**95.62**	**83.22**	85.49	89.32	**89.81**	85.53	**89.06**

Table 5: Ablation study on model design choices, including arbitrary shape modeling, different position encoding schemes, and token range settings. "Regular" denotes the average on *IIIT*, *SVT*, and *IC13* datasets,"Irregular" represents the average on *IC15*, *SVTP*, and *CUTE* datasets. Training settings are consistent with those in Tab. 1.

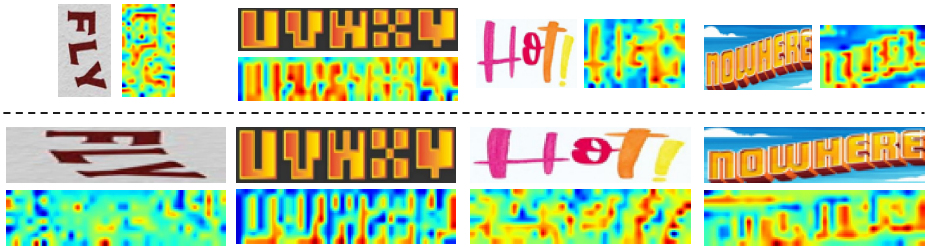

Fig. 4: Comparison of feature maps from different encoder outputs. **Top:** Images fed into the WATERec while preserving aspect ratio, and the corresponding encoder feature maps. **Bottom:** Baseline with fixed-template inputs (model in the first row of Tab. 5).

Model components. To understand the contribution of each component in WATERec, we perform ablations on arbitrary-shape modeling, positional encoding, and token range, as summarized in Tab. 5. Under arbitrary-shape modeling, adding a suitable positional encoding brings substantial improvements, and RoPE offers greater advantages compared with APE and SPE. Removing positional encoding (NoPE) entirely under arbitrary shapes leads to a large performance drop. This indicates that arbitrary-shape modeling provides the structural capacity to capture complex geometric deformations in artistic text, but its benefit is only fully realized when combined with an appropriate sequence modeling strategy. Expanding the range from a fixed $[256, 256]$ to a fully flexible $[1, 256]$ already yields noticeable improvements, as it allows the model to adapt sequence length to input complexity. However, very short sequences may be suboptimal for some small images. Setting the minimum limit of tokens to 64 (an example image size of 32×32) yields noticeable gains. Moreover, a larger maximum token limit can further improve text recognition accuracy. Nonetheless, to keep token consumption comparable with other methods, we adopt the range [64, 256] as our baseline. We further measure the effect of the token range on efficiency: when enlarging the range from $[64, 256]$ to $[64, 512]$, the FPS of WATERec drops from 361.66 to 191.46. Although a larger token budget can slightly improve accuracy (see Tab. 5), $[64, 256]$ provides a clearly better accuracy/efficiency trade-off.

5.4 More Visualization and Analysis

Multilingual Support As shown in Fig. 5, both subsets of WATER-S support multiple languages, and we extend them easily by replacing the corresponding language corpus.

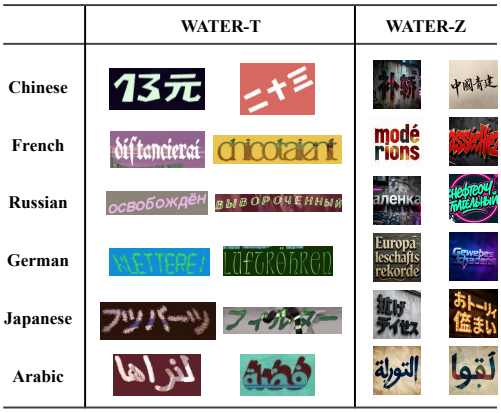

Fig. 5: Visualization of synthesized multilingual examples.

Arbitrary-shape modeling directs more robust attention. To intuitively verify the effectiveness of our arbitrary-shaped input design and to examine whether WATERec indeed robustly handles images with diverse shapes (especially artistic text), we visualize the feature maps produced by the WATERec encoder and by the vanilla ViT baseline with fixed-template inputs, as shown in Fig. 4. Benefiting from the preservation of the native aspect ratio, the WATERec encoder accurately and robustly extracts features from artistic text images with diverse shapes and layouts. In contrast, the vanilla ViT baseline stretches vertically oriented text, making the features less salient. This visualization further underscores the importance of supporting arbitrary-shaped inputs for STR (especially in WordArt).

Our strong baseline still has limitations. From the left of Fig. 6, we observe that WATERec consistently produces predictions matching the ground-truth, even under challenging conditions such as ambiguous characters, distorted fonts, multi-orientations, curved text, and perspective distortions. In contrast, existing STR models are more susceptible to these challenging scenarios and fail on some cases. This type of fine-grained confusion is even more pronounced for VLMs with small but critical errors that severely degrade the overall recognition quality. The right of Fig. 6 further presents several bad cases of WATERec. Most of these errors arise from inherent ambiguities from a semantic perspective, while some

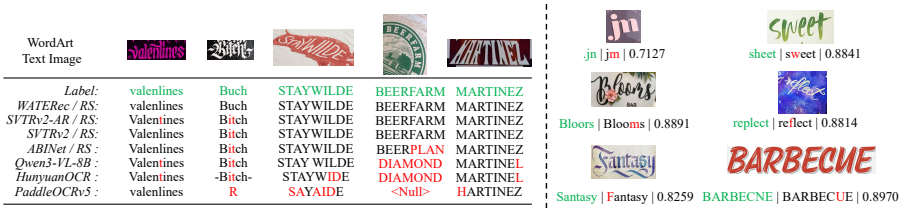

Fig. 6: Left: Visualization of different models' predictions on the WordArt-Bench, a supplement to Fig. 1. **Right:** Some bad-case examples from our final model (WATERec / RS), and the information below each image is formatted as: label | prediction | confidence. Characters differing from the ground truth are highlighted in red, and <Null> indicates that there is no output result.

predictions can even be considered reasonable (e.g., "Blooms", "BARBECUE"). We provide the complete set of bad-cases in the appendix, which offers a more comprehensive view of the current limitations of our approach.

6 Conclusion

We revisited artistic text recognition from both a data and model perspective, and showed that dedicated design is crucial for handling highly stylized, layout-rich WordArt scenarios. On the data side, we constructed **WATER-S**, a large-scale synthetic suite composed of one subset rendered with our SynthWordArt engine and the other subset from generative models, together with a carefully deduplicated real training set **WATER-R**. These datasets complement each other in font controllability, layout diversity, and visual realism, and consistently improve a wide range of STR models. On the model side, we proposed **WATERec**, which combines a NaViT-like encoder with RoPE and an autoregressive decoder. This design preserves native aspect ratios, mitigates distortion from fixed-template resizing, and better adapts to complex reading orders and artistic layouts. Experiments on WordArt-Bench, common STR benchmarks, and Union14M-Benchmark show that WATERec sets new advanced results, surpasses 90% accuracy on WordArt-Bench for the first time, and outperforms both general and OCR-specialized VLMs by a clear margin. Our results indicate that artistic text is still a challenging regime for current OCR and VLM systems, and that targeted synthetic data and arbitrary-shape modeling are effective tools for closing this gap. In future, we plan to extend our WATER pipeline to multilingual settings and to support more scripts in both benchmarks and synthetic data. We are also interested in improving VLMs for artistic text recognition through reasoning (e.g., chain-of-thought) to approach expert-model performance.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grants 625B2057 and 32341012.

References

1. Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., Ge, W., Guo, Z., Huang, Q., Huang, J., Huang, F., Hui, B., Jiang, S., Li, Z., Li, M., Li, M., Li, K., Lin, Z., Lin, J., Liu, X., Liu, J., Liu, C., Liu, Y., Liu, D., Liu, S., Lu, D., Luo, R., Lv, C., Men, R., Meng, L., Ren, X., Ren, X., Song, S., Sun, Y., Tang, J., Tu, J., Wan, J., Wang, P., Wang, P., Wang, Q., Wang, Y., Xie, T., Xu, Y., Xu, H., Xu, J., Yang, Z., Yang, M., Yang, J., Yang, A., Yu, B., Zhang, F., Zhang, H., Zhang, X., Zheng, B., Zhong, H., Zhou, J., Zhou, F., Zhou, J., Zhu, Y., Zhu, K.: Qwen3-vl technical report. CoRR **abs/2511.21631** (2025)
2. Bautista, D., Atienza, R.: Scene text recognition with permuted autoregressive sequence models. In: ECCV. pp. 178–196 (2022)
3. Cai, H., Cao, S., Du, R., Gao, P., Hoi, S., Hou, Z., Huang, S., Jiang, D., Jin, X., Li, L., et al.: Z-image: An efficient image generation foundation model with single-stream diffusion transformer. CoRR **abs/2511.22699** (2025)
4. Chen, J., Yu, H., Ma, J., Guan, M., Xu, X., Wang, X., Qu, S., Li, B., Xue, X.: Benchmarking chinese text recognition: Datasets, baselines, and an empirical study. CoRR **abs/2112.15093** (2021)
5. Cui, C., Sun, T., Liang, S., Gao, T., Zhang, Z., Liu, J., Wang, X., Zhou, C., Liu, H., Lin, M., et al.: Boosting document parsing efficiency and performance with coarse-to-fine visual processing. In: CVPR. pp. 16655–16665 (2026)
6. Cui, C., Sun, T., Lin, M., Gao, T., Zhang, Y., Liu, J., Wang, X., Zhang, Z., Zhou, C., Liu, H., et al.: Paddleocr 3.0 technical report. CoRR **abs/2507.05595** (2025)
7. Dehghani, M., Mustafa, B., Djolonga, J., Heek, J., Minderer, M., Caron, M., Steiner, A., Puigcerver, J., Geirhos, R., Alabdulmohsin, I.M., et al.: Patch n'pack: Navit, a vision transformer for any aspect ratio and resolution. NeurIPS **36**, 2252–2274 (2023)
8. Deshmukh, A.S., Chumachenko, K., Rintamaki, T., Le, M., Poon, T., Taheri, D.M., Karmanov, I., Liu, G., Seppanen, J., Chen, G., et al.: Nvidia nemotron nano v2 vl. CoRR **abs/2511.03929** (2025)
9. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021)
10. Du, Y., Chen, Z., Jia, C., Gao, X., Jiang, Y.G.: Out of length text recognition with sub-string matching. In: AAAI. vol. 39, pp. 2798–2806 (2025)
11. Du, Y., Chen, Z., Jia, C., Yin, X., Li, C., Du, Y., Jiang, Y.G.: Context perception parallel decoder for scene text recognition. IEEE Trans. Pattern Anal. Mach. Intell. **47**(6), 4668–4683 (2025). https://doi.org/10.1109/TPAMI.2025.3545453
12. Du, Y., Chen, Z., Jia, C., Yin, X., Zheng, T., Li, C., Du, Y., Jiang, Y.G.: SVTR: Scene text recognition with a single visual model. In: IJCAI. pp. 884–890 (2022)
13. Du, Y., Chen, Z., Xie, H., Jia, C., Jiang, Y.G.: Svtrv2: Ctc beats encoder-decoder models in scene text recognition. In: ICCV. pp. 20147–20156 (2025)
14. Du, Y., Chen, Z., Xie, Y., Bai, W., Feng, H., Shi, W., Su, Y., Huang, C., Jiang, Y.G.: Unirec-0.1b: Unified text and formula recognition with 0.1b parameters. arXiv preprint arXiv:2512.21095 (2025)
15. Du, Y., Chen, Z., Yuchen, S., Jia, C., Jiang, Y.G.: Instruction-guided scene text recognition. IEEE Trans. Pattern Anal. Mach. Intell. **47**(4), 2723–2738 (2025)
16. Du, Y., Zhao, M., Fan, S., Chen, Z., Jia, C., Jiang, Y.G.: Mdiff4str: Mask diffusion model for scene text recognition. In: AAAI. vol. 40, pp. 3705–3713 (2026)

17. Fang, S., Xie, H., Wang, Y., Mao, Z., Zhang, Y.: Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition. In: CVPR. pp. 7098–7107 (2021)
18. Graves, A., Fernández, S., Gomez, F., Schmidhuber, J.: Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In: ICML. pp. 369–376 (2006)
19. Guan, T., Shen, W., Yang, X.: Ccdplus: Towards accurate character to character distillation for text recognition. IEEE Trans. Pattern Anal. Mach. Intell. **47**(5), 3546–3562 (2025). https://doi.org/10.1109/TPAMI.2025.3533737
20. Guan, T., Shen, W., Yang, X., Feng, Q., Jiang, Z., Yang, X.: Self-supervised character-to-character distillation for text recognition. In: ICCV. pp. 19473–19484 (2023)
21. Gupta, A., Vedaldi, A., Zisserman, A.: Synthetic data for text localisation in natural images. In: CVPR. pp. 2315–2324 (2016)
22. Heo, B., Park, S., Han, D., Yun, S.: Rotary position embedding for vision transformer. In: ECCV. pp. 289–305 (2024)
23. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al.: Lora: Low-rank adaptation of large language models. In: ICLR (2022)
24. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Synthetic data and artificial neural networks for natural scene text recognition. CoRR **abs/1406.2227** (2014)
25. Jiang, Q., Wang, J., Peng, D., Liu, C., Jin, L.: Revisiting scene text recognition: A data perspective. In: ICCV. pp. 20486–20497 (2023)
26. Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanov, A., Iwamura, M., Matas, J., Neumann, L., Chandrasekhar, V.R., Lu, S., et al.: ICDAR 2015 competition on robust reading. In: ICDAR. pp. 1156–1160 (2015)
27. Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., i Bigorda, L.G., Mestre, S.R., Mas, J., Mota, D.F., Almazan, J.A., De Las Heras, L.P.: ICDAR 2013 robust reading competition. In: ICDAR. pp. 1484–1493 (2013)
28. Labs, B.F., Batifol, S., Blattmann, A., Boesel, F., Consul, S., Diagne, C., Dockhorn, T., English, J., English, Z., Esser, P., Kulal, S., Lacey, K., Levi, Y., Li, C., Lorenz, D., Müller, J., Podell, D., Rombach, R., Saini, H., Sauer, A., Smith, L.: Flux.1 kontext: Flow matching for in-context image generation and editing in latent space. CoRR **abs/2506.15742** (2025)
29. Li, H., Wang, P., Shen, C., Zhang, G.: Show, attend and read: A simple and strong baseline for irregular text recognition. In: AAAI. vol. 33, pp. 8610–8617 (2019)
30. Li, M., Lv, T., Chen, J., Cui, L., Lu, Y., Florencio, D., Zhang, C., Li, Z., Wei, F.: Trocr: Transformer-based optical character recognition with pre-trained models. In: AAAI. vol. 37, pp. 13094–13102 (2023)
31. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: ICLR (2019)
32. Mishra, A., Alahari, K., Jawahar, C.: Scene text recognition using higher order language priors. In: BMVC. pp. 1–11 (2012)
33. Phan, T.Q., Shivakumara, P., Tian, S., Tan, C.L.: Recognizing text with perspective distortion in natural scenes. In: ICCV. pp. 569–576 (2013)
34. Qiao, Z., Zhou, Y., Wei, J., Wang, W., Zhang, Y., Jiang, N., Wang, H., Wang, W.: Pimnet: a parallel, iterative and mimicking network for scene text recognition. In: ACM MM. pp. 2046–2055 (2021)
35. Qiao, Z., Zhou, Y., Yang, D., Zhou, Y., Wang, W.: Seed: Semantics enhanced encoder-decoder framework for scene text recognition. In: CVPR. pp. 13528–13537 (2020)

36. Risnumawan, A., Shivakumara, P., Chan, C.S., Tan, C.L.: A robust arbitrary text detection system for natural scene images. ESWA **41**(18), 8027–8048 (2014)
37. Sheng, F., Chen, Z., Xu, B.: NRTR: A no-recurrence sequence-to-sequence model for scene text recognition. In: ICDAR. pp. 781–786 (2019)
38. Shi, B., Bai, X., Yao, C.: An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE Trans. Pattern Anal. Mach. Intell. **39**(11), 2298–2304 (2016). https://doi.org/10.1109/TPAMI.2016.2646371
39. Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., Liu, Y.: Roformer: Enhanced transformer with rotary position embedding. Neurocomputing **568**, 127063 (2024)
40. Team, H.V., Lyu, P., Wan, X., Li, G., Peng, S., Wang, W., Wu, L., Shen, H., Zhou, Y., Tang, C., et al.: Hunyuanocr technical report. CoRR **abs/2511.19575** (2025)
41. Wang, K., Babenko, B., Belongie, S.: End-to-end scene text recognition. In: ICCV. pp. 1457–1464 (2011)
42. Wang, W., Gao, Z., Gu, L., Pu, H., Cui, L., Wei, X., Liu, Z., Jing, L., Ye, S., Shao, J., et al.: Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. CoRR **abs/2508.18265** (2025)
43. Wei, H., Liu, C., Chen, J., Wang, J., Kong, L., Xu, Y., Ge, Z., Zhao, L., Sun, J., Peng, Y., et al.: General ocr theory: Towards ocr-2.0 via a unified end-to-end model. CoRR **abs/2409.01704** (2024)
44. Wei, H., Sun, Y., Li, Y.: Deepseek-ocr: Contexts optical compression. CoRR **abs/2510.18234** (2025)
45. Wei, H., Sun, Y., Li, Y.: Deepseek-ocr 2: Visual causal flow. CoRR **abs/2601.20552** (2026)
46. Wei, J., Zhan, H., Lu, Y., Tu, X., Yin, B., Liu, C., Pal, U.: Image as a language: Revisiting scene text recognition via balanced, unified and synchronized vision-language reasoning network. In: AAAI. vol. 38, pp. 5885–5893 (2024)
47. Wu, C., Li, J., Zhou, J., Lin, J., Gao, K., Yan, K., Yin, S.m., Bai, S., Xu, X., Chen, Y., et al.: Qwen-image technical report. CoRR **abs/2508.02324** (2025)
48. Xie, X., Deng, L., Zhang, Z., Wang, Z., Liu, Y.: Icdar 2024 competition on artistic text recognition. In: ICDAR. pp. 301–314 (2024)
49. Xie, X., Fu, L., Zhang, Z., Wang, Z., Bai, X.: Toward understanding wordart: Corner-guided transformer for scene text recognition. In: ECCV. pp. 303–321 (2022)
50. Xie, X., Li, Y., Liu, Y., Zhang, Z., Wang, Z., Xiong, W., Bai, X.: Was: Dataset and methods for artistic text segmentation. In: ECCV. pp. 237–254 (2024)
51. Xu, J., Wang, Y., Xie, H., Zhang, Y.: Ote: Exploring accurate scene text recognition using one token. In: CVPR. pp. 28327–28336 (2024)
52. Xu, X., Zhang, Z., Wang, Z., Price, B., Wang, Z., Shi, H.: Rethinking text segmentation: A novel dataset and a text-specific refinement approach. In: CVPR. pp. 12045–12055 (2021)
53. Yang, X., Qiao, Z., Zhou, Y.: Ipad: iterative, parallel, and diffusion-based network for scene text recognition. Int. J. Comput. Vis. **133**(8), 5589–5609 (2025)
54. Ye, X., Du, Y., Tao, Y., Chen, Z.: Textssr: Diffusion-based data synthesis for scene text recognition. In: ICCV. pp. 17464–17473 (2025)
55. Ye, X., Du, Y., Zhang, J., Li, C., LYU, J., Chen, Z.: What's wrong with synthetic data for scene text recognition? a strong synthetic engine with diverse simulations and self-evolution. In: CVPR. pp. 16645–16654 (2026)
56. Yim, M., Kim, Y., Cho, H.C., Park, S.: Synthtiger: Synthetic text image generator towards better text recognition models. In: ICDAR. pp. 109–124 (2021)

57. Yu, D., Li, X., Zhang, C., Liu, T., Han, J., Liu, J., Ding, E.: Towards accurate scene text recognition with semantic reasoning networks. In: CVPR. pp. 12113–12122 (2020)
58. Zhai, C., Chen, Z., Li, J., Xu, B.: Chinese image text recognition with blstm-ctc: a segmentation-free method. In: CCPR. pp. 525–536 (2016)
59. Zhang, B., Xie, H., Wang, Y., Xu, J., Zhang, Y.: Linguistic more: Taking a further step toward efficient and accurate scene text recognition. In: IJCAI. pp. 1704–1712 (2023)
60. Zheng, T., Chen, Z., Fang, S., Xie, H., Jiang, Y.G.: Cdistnet: Perceiving multi-domain character distance for robust text recognition. Int. J. Comput. Vis. **132**(2), 300–318 (2024)
61. Zhu, Y., Liu, J., Gao, F., Liu, W., Wang, X., Wang, P., Huang, F., Yao, C., Yang, Z.: Visual text generation in the wild. In: ECCV. pp. 89–106 (2024)