INKFLOW: BEYOND STYLE TRANSFER VIA WRITING FLOW LEARNING FOR ONLINE HANDWRITING GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Stylized handwriting generation aims to synthesize new content with specific writer characteristics from reference handwriting samples. Existing methods typically follow the style transfer paradigm, treating style features disentangled from reference samples as the transferable object and fusing them with target text to generate the final result. This paradigm focuses on the "style presented in the outcome." In contrast, we focus on "how such outcomes are produced": a specific handwriting result can be viewed as a manifestation of how a writer transforms textual content into trajectories under a given content condition. Accordingly, we redefine the transferable object from static style features to the underlying content-to-trajectory transformation, which we term the **writing flow**, and transfer it to new content generation. Based on this perspective, we propose **InkFlow**, a writing-flow learning framework for online handwriting generation. Given a complete reference text–trajectory example, InkFlow models the underlying content-to-trajectory relationship and transfers it to new target text. Specifically, we adopt a latent diffusion framework with a task-aware VAE that learns a structured handwriting latent space with both content- and style-aware properties, and design a reference–target conditioned Diffusion Transformer to implicitly model and transfer writing flows. Furthermore, we introduce a writing-flow alignment objective to enhance the preservation of reference writing characteristics in the generated results. Extensive experiments on the IAM-OnDB and ICDAR-2013 text-line handwriting benchmarks demonstrate that our model achieves strong performance in both content fidelity and writer-style consistency. Code will be publicly available.

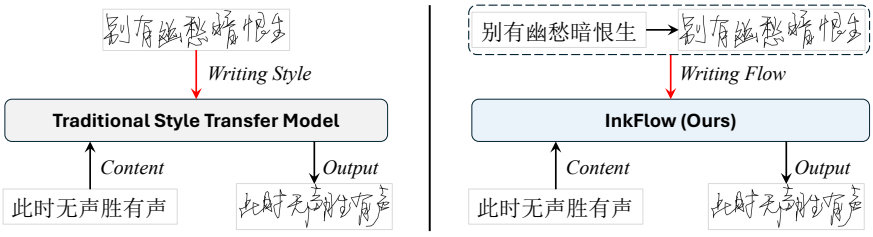


Figure 1: **Beyond style transfer for stylized handwriting generation.** Traditional methods typically rely on an additional style encoder to extract and transfer explicit style representations from reference handwriting. In contrast, **InkFlow (Ours)** models the transferable content-to-trajectory transformation, termed the **writing flow**, directly within a unified latent space for personalized generation of new content.

1 INTRODUCTION

Stylized handwriting generation (SHG) aims to synthesize new handwriting with specific writer characteristics from reference samples, with applications in personalized digital writing, document generation, and human–computer interaction. Unlike general image generation, online handwriting generation must jointly preserve textual content and writer-specific writing characteristics, including stroke organization, trajectory dynamics, and writing habits. Learning transferable handwriting

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

characteristics from limited reference samples, while preserving coherent structure and writing dynamics over long text lines, therefore remains a central challenge in SHG.

Existing methods typically formulate this problem under the style transfer paradigm, where handwriting style is treated as an attribute that can be relatively disentangled from content, extracted from reference samples, and transferred to new content. Following this formulation, previous studies (Dai et al., 2023; Ren et al., 2025; 2023; Liu et al., 2024; Tang & Lian, 2021; Chang et al., 2022; Pan et al., 2026a; Wang et al., 2026a) mainly focus on two aspects. First, they design robust style modeling modules to capture writer-specific characteristics from limited references, including global and local patterns, multi-scale structures, or different frequency components. Second, they develop effective content–style interaction and decoding mechanisms, such as feature fusion and attention-based interaction, together with auxiliary objectives that encourage content preservation and style consistency. As illustrated on the left side of Fig. 1, these methods ultimately compress the observed reference handwriting into explicit style representations and inject them into target generation.

While effective, this paradigm primarily transfers the style characteristics observed in limited reference samples, emphasizing **what the handwriting looks like rather than how it is generated**. Such static representations do not directly model the transformation from textual content to handwriting trajectories, which may limit their ability to generalize across unseen contents, long text lines, and diverse writing systems.

In this work, we rethink SHG from the perspective of transferable generation mechanisms. We view handwriting style not merely as an observable attribute of the final handwriting result, but as the dynamic generation pattern through which a writer transforms content into trajectories. From this perspective, the final handwriting is only a manifestation of the underlying generation process under a specific content condition, while the transferable information lies in the content-to-trajectory transformation. We refer to this writer-specific transformation pattern as the **writing flow**, which characterizes how textual content is progressively realized as handwriting trajectories. As illustrated on the right side of Fig. 1, we therefore reformulate stylized handwriting generation as a writing flow learning problem, aiming to learn such transferable transformation patterns from handwriting demonstrations rather than extracting isolated style representations from limited samples.

Based on this perspective, we propose **InkFlow**, a writing flow learning framework for online handwriting generation that learns transferable content-to-trajectory transformations directly from handwriting demonstrations. Unlike conventional approaches that compress reference handwriting into static style representations, InkFlow exploits the correspondence between the textual content and trajectory of a complete reference demonstration to establish and transfer writing flows. Specifically, we first construct a task-aware VAE latent space with content- and style-aware regularization, and perform writing flow learning entirely within this latent space. Built upon it, a reference–target conditioned Diffusion Transformer jointly models the reference content, reference trajectory, and target content to implicitly establish and transfer writing flows. Moreover, the style-aware structure already learned by the VAE provides a readily available perceptual space for **Writing-Flow Alignment**, enabling reference-aware regularization without introducing additional learnable modules. Together, these designs enable InkFlow to learn and transfer writing flows from a single complete handwriting demonstration across diverse contents and writing systems. Extensive experiments on multiple online handwriting generation benchmarks demonstrate that InkFlow achieves strong performance in both content fidelity and writer consistency.

The main contributions of this work are summarized as follows: 1) We introduce **writing flow learning** as an alternative formulation for stylized handwriting generation, shifting the transferable target from static style representations to writer-specific content-to-trajectory transformations. 2) We propose **InkFlow**, a latent diffusion framework that combines a task-aware VAE, a reference–target conditioned Diffusion Transformer, and writing-flow alignment to learn and transfer writing flows from a single handwriting demonstration. 3) We evaluate InkFlow on Chinese and English text-line online handwriting generation benchmarks, where it achieves strong performance in both content fidelity and writer consistency.

2 Related Work

Stylized Character Generation. Stylized generation has been widely studied in both font synthesis and handwriting generation. For font generation, existing methods typically learn character appearance from a small number of style references and transfer the extracted style to unseen char-

acters (Pan et al., 2023; Zhu et al., 2026; Park et al., 2024; Liu & Lian, 2024; Li & Lian, 2024; Chen et al., 2025; Wang et al., 2023; Cai et al., 2026; Yang et al., 2024). For handwriting generation, a similar formulation has been adopted to synthesize writer-specific handwriting from reference samples (Dai et al., 2023; Tang & Lian, 2021; Li et al., 2024; Liu et al., 2024; Pippi et al., 2025; Ren et al., 2025; Dai et al., 2024; Ren et al., 2023; Tang et al., 2019; Wang et al., 2026c; Dai et al., 2025). Despite differences in architecture and modality, most of these methods follow a common style-transfer paradigm, where content and style are treated as separable factors: style is explicitly encoded from reference samples and subsequently fused with content representations through feature fusion, cross-attention, or adaptive normalization.

This formulation generally assumes that writer style can be represented as an independent transferable factor extracted from reference handwriting. In contrast, InkFlow focuses on modeling the content-to-trajectory transformation itself as the transferable writing flow.

Diffusion Generative Models. Diffusion models (Ho et al., 2020) have achieved strong generative performance across a wide range of modalities. To improve efficiency, latent diffusion models (Rombach et al., 2022) perform the denoising process in a compact representation space learned by a variational autoencoder, while Diffusion Transformers (Peebles & Xie, 2023) replace convolutional U-Net backbones with Transformer architectures to enable more flexible global interactions and conditional modeling. These developments have also motivated diffusion-based handwriting generation methods (Ren et al., 2023; 2025; Dai et al., 2025; Pan et al., 2026a;b).

InkFlow builds upon the latent diffusion and DiT paradigms, but focuses on reference-conditioned sequence generation. Rather than injecting a separately encoded style feature into the denoising network, our DiT jointly models reference textual content, reference handwriting trajectories, and target content, allowing writing-flow establishment and transfer to emerge through structured reference–target interactions in the latent space.

3 METHOD

3.1 PROBLEM FORMULATION

Online Handwriting Data. An online handwriting text-line sample is represented as a pair (T, X), where $T = \{c_1, c_2, \ldots, c_k\}$ denotes a sequence of k characters, and $X = \{s_1, s_2, \ldots, s_k\}$ is the corresponding sequence of pen trajectories. Each $s_i \in \mathbb{R}^{l_i \times 4}$ encodes the trajectory of character c_i with l_i sampled points. The total number of points in the entire text line is $\sum_{i=1}^{k} l_i$. Each point is represented as a 4-dimensional vector (x, y, pen), where (x, y) denote the spatial coordinates and pen is a one-hot vector indicating the pen state: *Pen Down* or *Pen Up*. Examples of handwriting samples and their corresponding trajectory representations are provided in Appendix A.1.

3.2 FROM STYLE TRANSFER TO WRITING FLOW LEARNING

Task Definition. Given target text content T and reference information $\mathcal{R}$, which represents the distinctive writing characteristics demonstrated by the reference handwriting samples, SHG aims to generate a handwriting trajectory X that follows the target content while preserving the referenced writing characteristics. Different formulations mainly differ in how $\mathcal{R}$ is represented.

$$T \xrightarrow{\mathcal{R}} X. \tag{1}$$

Traditional Style Transfer Formulation. In conventional style transfer methods (Dai et al., 2023; 2024; Ren et al., 2023; 2025; Tang & Lian, 2021; Luo et al., 2022), the reference information $\mathcal{R}$ in the handwriting generation objective is typically represented as an explicit style feature extracted from multiple reference handwriting samples. Given a set of reference samples X_{ref}, a style encoder $E_s(\cdot)$ maps the observed handwriting into a compact style representation:

$$\mathcal{R}_{\mathrm{sty}} = E_s(X_{\mathrm{ref}}). \tag{2}$$

The resulting representation is then combined with the target content to guide handwriting generation. In this formulation, the transferable information is mainly derived from the observed handwriting results themselves, while the relationship between the reference contents and their corresponding trajectories is not explicitly considered.

Our Writing Flow Formulation. The style transfer formulation above relies on the assumption that handwriting content and writer style can be relatively disentangled, with the transferable information summarized as a compact style representation extracted from observed handwriting. Under this formulation, what can be transferred is largely determined by what such a representation can capture from the available reference samples.

Recent studies have begun to reconsider the form of transferable information, exploring richer content–style interactions as well as generation or transfer processes beyond a single static style representation (Chang et al., 2022; Fu et al., 2024; 2025; Wang et al., 2026b; Shin et al., 2026). Building on this direction, we further ask what remains consistent when the same writer produces different contents, and therefore what should be transferred from one handwriting demonstration to another. Rather than treating style as an isolated appearance descriptor, we view this transferable factor as the writer-specific transformation that governs how textual content is realized as handwriting trajectories.

Specifically, for a reference handwriting sample X_{ref}, the corresponding textual content T_{ref} can be obtained from dataset annotations or estimated using an OCR model. The resulting content–trajectory correspondence provides an observation of how the writer realizes textual content as handwriting trajectories. We refer to the transferable regularity underlying this correspondence as the **writing flow**, and formulate the reference information as

$$\mathcal{R}_{\text{flow}} = \mathcal{F}_w = \Phi(T_{\text{ref}}, X_{\text{ref}}), \tag{3}$$

where $\mathcal{F}_w$ characterizes the writer-specific content-to-trajectory transformation demonstrated by the reference sample, and $\Phi(\cdot)$ denotes this relationship conceptually rather than an explicit feature encoder. Accordingly, the transferable reference information is no longer defined solely by an independently extracted style feature, but by the correspondence between *what is written* and *how it is written*. This writing flow can then serve as the reference information $\mathcal{R}$ for generating new content.

3.3 INKFLOW FRAMEWORK

Based on the writing flow formulation, we introduce **InkFlow**, a latent diffusion framework that learns transferable writing flows from handwriting demonstrations. As illustrated in Fig. 3, InkFlow models writing flow through the joint interaction between reference content, reference trajectory, and target content in a unified latent space, rather than explicitly encoding handwriting style as an independent representation. The reference pair $(T_{\text{ref}}, X_{\text{ref}})$ provides an observation of how content is transformed into handwriting, while T_{gen} specifies the new content to be generated.

InkFlow consists of two components: (1) a task-aware variational autoencoder that learns a compact handwriting latent space, and (2) a Writing Flow DiT that establishes transferable writing flows from reference content–trajectory correspondences and applies them to target generation.

3.3.1 TASK-AWARE LATENT SPACE LEARNING

To provide a compact shared feature space for subsequent writing flow modeling, InkFlow employs a Transformer-based variational autoencoder to encode online handwriting trajectories into latent representations. Following DiffInk (Pan et al., 2026a), the VAE reconstructs trajectories in a non-autoregressive manner, where continuous pen coordinates are modeled with a Gaussian mixture model (Reynolds, 2015) and discrete pen states are optimized with a classification loss. The reconstruction objective $\mathcal{L}_{\text{rec}}$ further includes KL regularization to maintain a compact and smooth latent space.

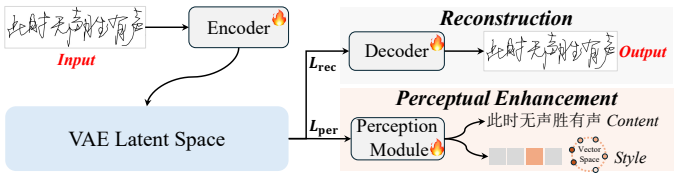


Figure 2: **Task-aware latent space learning.** A VAE is trained with trajectory reconstruction and perceptual enhancement objectives to learn a structured handwriting latent space.

Since reconstruction fidelity alone does not necessarily yield representations well suited for downstream generation, we further introduce perceptual enhancement to improve the content and handwriting awareness of the latent space. Specifically, a lightweight text recognition head $\mathcal{H}_{\text{ctc}}$ employs

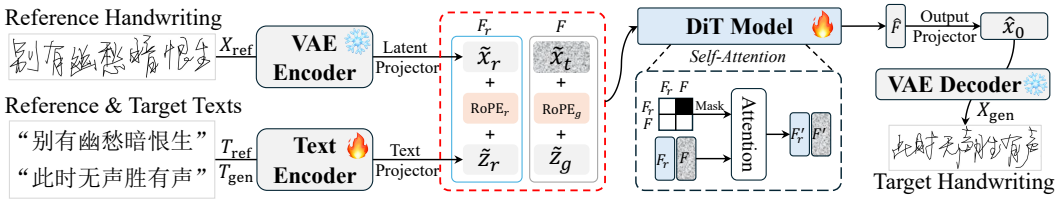


Figure 3: **InkFlow (Ours)** forms the reference writing-flow representation F_r from the features of the paired reference content T_{ref} and trajectory X_{ref}, and constructs the target generation flow F from the target content T_{gen} and noisy handwriting latent. Their interaction in a unified feature space enables writing-flow transfer for personalized handwriting generation.

CTC (Graves et al., 2006) supervision to preserve character-level structural information, while a writer discrimination head $\mathcal{H}_{\text{sty}}$ captures writer-specific handwriting variations. Following (Pan et al., 2026a), writer classification provides style-aware supervision; we further introduce center and triplet losses on $\mathcal{H}_{\text{sty}}$ to enhance intra-writer compactness and inter-writer separability. These auxiliary objectives are jointly summarized as the perceptual enhancement loss $\mathcal{L}_{\text{per}}$. The overall VAE objective is formulated as

$$\mathcal{L}_{\text{VAE}} = \mathcal{L}_{\text{rec}} + \lambda_{\text{per}}\mathcal{L}_{\text{per}}. \tag{4}$$

The resulting task-aware latent space, together with $\mathcal{H}_{\text{ctc}}$ and $\mathcal{H}_{\text{sty}}$, is then kept fixed for subsequent writing flow learning. The detailed VAE architecture can be found in Appendix A.2.

3.3.2 Writing Flow Learning with Diffusion Transformer

Built upon the task-aware latent space, InkFlow employs a Diffusion Transformer (DiT) (Peebles & Xie, 2023) to learn transferable writing flows without explicitly extracting an independent style representation. Given a reference demonstration $(T_{\text{ref}}, X_{\text{ref}})$ and target content T_{gen}, Writing Flow DiT jointly models the reference content–trajectory correspondence and transfers this relationship to target generation.

Writing Flow Representation. Following joint and in-context generative modeling paradigms that place conditioning information and generation states in a shared Transformer space (Bao et al., 2023; Cao et al., 2023), we represent the reference demonstration and target generation within a unified feature space. For textual content, T_{ref} and T_{gen} are embedded at the character level and padded with a shared learnable token to match their corresponding handwriting latent lengths. The resulting sequences are processed by a lightweight ConvNeXt-V2 (Woo et al., 2023) encoder and projected into the DiT hidden space, yielding the reference and target text features $\widetilde{z}_r$ and $\widetilde{z}_g$. For handwriting, the frozen VAE encodes X_{ref} into the reference latent x_r, while the target side uses the noisy latent x_t at diffusion timestep t; both are projected into the same hidden space as $\widetilde{x}_r$ and $\widetilde{x}_t$. We then combine the textual and handwriting features to construct the reference writing-flow representation F_r and the target generation-flow representation F:

$$F_r = \widetilde{z}_r + \widetilde{x}_r + \text{RoPE}_r, \qquad F = \widetilde{z}_g + \widetilde{x}_t + \text{RoPE}_g. \tag{5}$$

Here, F_r couples *what is written* with *how it is written*, thereby representing the reference-side writing flow, while F couples the target content with its current noisy handwriting state and serves as the generation flow. The RoPE coordinates (Su et al., 2024) of the two streams are independently initialized from zero to preserve their respective temporal structures. Finally, F_r and F are concatenated as $F_{\text{all}} = [F_r; F]$ for joint modeling in Writing Flow DiT.

Writing Flow Modeling and Transfer. A standard self-attention operation would permit full bidirectional interaction between the clean reference stream and the evolving noisy target stream. Since F_r is intended to preserve a stable reference-side writing-flow representation, allowing the noisy target state to update it may interfere with the underlying reference content–trajectory correspondence. Following structured attention designs that regulate directional information flow between conditioning and prediction tokens (Cao et al., 2024), we therefore introduce an asymmetric Writing-Flow Mask:

$$\text{SelfAttn}(F_{\text{all}}; \mathbf{M}), \qquad \mathbf{M} = \begin{bmatrix} \mathbf{1}_{rr} & \mathbf{0}_{rg} \\ \mathbf{1}_{gr} & \mathbf{1}_{gg} \end{bmatrix}, \quad \mathbf{1}: \text{ attend}, \ \mathbf{0}: \text{ mask}. \tag{6}$$

Here, rows correspond to queries and columns to keys and values. The mask preserves reference-to-reference, generation-to-reference, and generation-to-generation interactions, while blocking reference-to-generation attention. In this way, F_r establishes the reference content-to-trajectory correspondence, while F accesses and transfers the resulting writing flow during denoising without perturbing the reference representation.

After processing F_{all} with Writing Flow DiT, we retain the output features corresponding to the generation representation F and project them through a linear prediction head back to the VAE latent space, yielding the predicted clean latent $\hat{x}_0$.

Diffusion Optimization Objective. Given the established writing flow, Writing Flow DiT performs conditional diffusion learning in the handwriting latent space. Let x_0 denote the clean target latent encoded by the frozen VAE. At diffusion timestep t, Gaussian noise is added according to

$$x_t = \sqrt{\bar{\alpha}_t}\, x_0 + \sqrt{1-\bar{\alpha}_t}\, \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I). \tag{7}$$

The model predicts the clean latent $\hat{x}_0$ from the noisy target latent under the reference and target conditions. Following DiffInk (Pan et al., 2026a), we adopt the x_0-prediction formulation and optimize a token-level reconstruction objective for diffusion denoising:

$$\mathcal{L}_{\text{tok}} = \mathbb{E}_{t,\epsilon}\left[\|x_0 - \hat{x}_0\|_2^2\right]. \tag{8}$$

Although the token-level objective provides dense supervision for latent reconstruction, the reference condition is optimized only indirectly through the reconstruction error. Similar limitations have motivated prior conditional diffusion methods to introduce additional perceptual, contrastive, or guidance-based objectives for explicitly strengthening condition adherence (Cao et al., 2024; Yang et al., 2023; Luo et al., 2024; Shen et al., 2024; Mayr et al., 2025). In particular, uniformly averaging token-wise reconstruction errors does not explicitly emphasize the reference-dependent writing characteristics shared across the entire handwriting sequence. This issue is especially relevant to InkFlow, where such characteristics are not injected as an explicit style representation but must be inferred and transferred from the reference stream.

To provide more direct supervision for reference utilization, we introduce a **Writing-Flow Alignment** regularization by reusing the frozen style-aware head $\mathcal{H}_{\text{sty}}$ learned during VAE training, providing an additional perceptual constraint without introducing extra learnable modules or annotations. Specifically, we compute $\mathbf{s}^{\text{pred}} = \mathcal{H}_{\text{sty}}(\hat{x}_0)$ and $\mathbf{s}^{\text{ref}} = \mathcal{H}_{\text{sty}}(x_r)$, and optimize a reference-anchored supervised contrastive objective (Khosla et al., 2020; Yang et al., 2023):

$$\mathcal{L}_{\text{flow}} = \mathcal{L}_{\text{con}}\left(\mathbf{s}^{\text{pred}}, \mathbf{s}^{\text{ref}}, y\right), \tag{9}$$

where y denotes the writer identity. This objective pulls predictions toward references from the same writer while separating those from different writers, thereby strengthening the transfer of reference writing characteristics. The overall diffusion objective is defined as:

$$\mathcal{L}_{\text{diff}} = \mathcal{L}_{\text{tok}} + \lambda_{\text{flow}}\mathcal{L}_{\text{flow}}. \tag{10}$$

Classifier-free Guidance. Following multi-condition CFG (Chen et al., 2024; Pan et al., 2026a; Ho & Salimans, 2022), we use two independent condition-drop strategies during training: dropping the reference demonstration with probability p_r, or dropping both the reference and target-text conditions with probability p_u. During inference, target-content and reference-writing-flow guidance are controlled separately by w_c and w_r, respectively. DDIM (Song et al., 2020) sampling is used for generation. Further details of the Writing Flow DiT are provided in Appendix A.3.

4 Experiments

4.1 Dataset & Evaluation Metrics

We evaluate InkFlow on both Chinese and English online handwritten text-line datasets. For Chinese handwriting, we train InkFlow on CASIA-OLHWDB 2.0–2.2 (Liu et al., 2011), using 815 writers and approximately 39,969 text lines for training. The remaining 204 writers with approximately 10,061 text lines are used as the test set. We additionally evaluate the model on the ICDAR-2013

Table 1: Quantitative comparison of InkFlow with existing methods for Chinese handwritten text-line generation on the OLHWDB-2.0-2.2 benchmark.

End-to-End Text-line Generation							Character-level Generation with Real Layout					
Method	**Venue**	**AR%↑**	**CR%↑**	**FID↓**	**HWD↓**	**User↑**	**Method**	**Venue**	**AR%↑**	**CR%↑**	**FID↓**	**HWD↓**
OLHWG	ICLR'25	91.95	92.61	8.18	0.67	18.21	WriteLikeU	CGF'23	93.26	94.03	7.12	0.63
DiffInk	ICLR'26	93.75	94.14	5.42	0.43	32.76	SDT	CVPR'23	98.31	98.98	7.47	0.66
InkFlow	**Ours**	**95.78**	**96.02**	**2.30**	**0.29**	**49.05**	EleWritten	ECCV'24	93.93	94.50	11.77	0.70

text-line test set (Yin et al., 2013), which contains 3,370 text lines from 60 writers. For English handwriting, we use IAM-OnDB (Liwicki & Bunke, 2005), with 200 writers and 8,742 text lines for training, and the remaining 28 writers and 1,251 text lines for evaluation.

We evaluate the generated handwriting in the offline image domain to reduce biases caused by differences in online sequence preprocessing across methods. For content fidelity, following (Pan et al., 2026a; Dai et al., 2025; Yin et al., 2013), we report character accuracy rate (AR) and character correct rate (CR) for Chinese handwriting, and character error rate (CER) and word error rate (WER) for English handwriting. For generation quality and handwriting fidelity, we adopt Fréchet Inception Distance (FID Heusel et al. (2017)) and Handwriting Distance (HWD Pippi et al. (2023)) to measure distribution-level similarity and handwriting-style consistency, respectively. In addition to these quantitative metrics, following Wang et al. (2026b), we conduct a user study to provide complementary human perceptual evaluation of the generated handwriting.

4.2 Implementation Details

The VAE downsamples handwriting trajectories by a factor of 8 and is trained for 340K iterations using AdamW (Loshchilov & Hutter, 2017) with a learning rate of 5×10^{-4}. The reconstruction and perceptual loss weights are set to 1.0 and $\lambda_{\text{per}} = 0.1$, respectively. Writing Flow DiT contains 18 Transformer blocks with a hidden dimension of 896 and is trained for 250K iterations using AdamW with a learning rate of 7.5×10^{-5}, and $\lambda_{\text{flow}} = 0.005$. We use 1,000 diffusion timesteps with a cosine noise schedule and 20-step DDIM sampling for inference. The condition-drop probabilities are $p_r = 0.2$ and $p_u = 0.1$, the guidance scales are $w_c = 2.0$ and $w_r = 1.5$.

4.3 Main Results

For Chinese handwritten text-line generation, we compare InkFlow with WriteLikeU (Tang & Lian, 2021), SDT (Dai et al., 2023), EleWritten (Liu et al., 2024), OLHWG (Ren et al., 2025), and DiffInk (Pan et al., 2026a). For methods originally developed for isolated-character generation, we independently generate characters and reconstruct text lines using ground-truth character bounding boxes to provide a consistent layout protocol.

Table 2: Quantitative comparison on the Chinese ICDAR-2013 benchmark.

Method	**AR%↑**	**CR%↑**	**FID↓**	**HWD↓**
DiffInk (ICLR'26)	93.79	94.09	4.64	0.34
InkFlow (Ours)	**94.76**	**95.05**	**3.50**	**0.28**

As shown in Tab. 1, InkFlow achieves strong overall performance among end-to-end text-line generation methods. In terms of content accuracy, it improves AR/CR over DiffInk by 2.03/1.88 percentage points and over OLHWG by 3.83/3.41 percentage points, while also achieving lower FID and HWD. The qualitative results in Fig. 4 and Fig. 10 further show more natural text-line structures and stronger consistency with the reference handwriting. Among character-level methods, SDT achieves the highest content accuracy. However, its FID and HWD remain worse than those of InkFlow. More generally, WriteLikeU, SDT, and EleWritten all obtain higher FID and HWD than the end-to-end methods DiffInk and InkFlow, suggesting that independently generated characters, even when composed with real layout information, still exhibit gap from natural text-line handwriting.

We further evaluate InkFlow on ICDAR-2013. Since character-level bounding-box annotations are unavailable, character-based baselines cannot be evaluated under the same composition protocol, and we therefore compare with the end-to-end method DiffInk. As shown in Tab. 2, InkFlow improves AR/CR by 0.97/0.96 percentage points over DiffInk, while reducing FID/HWD by 1.14/0.06.

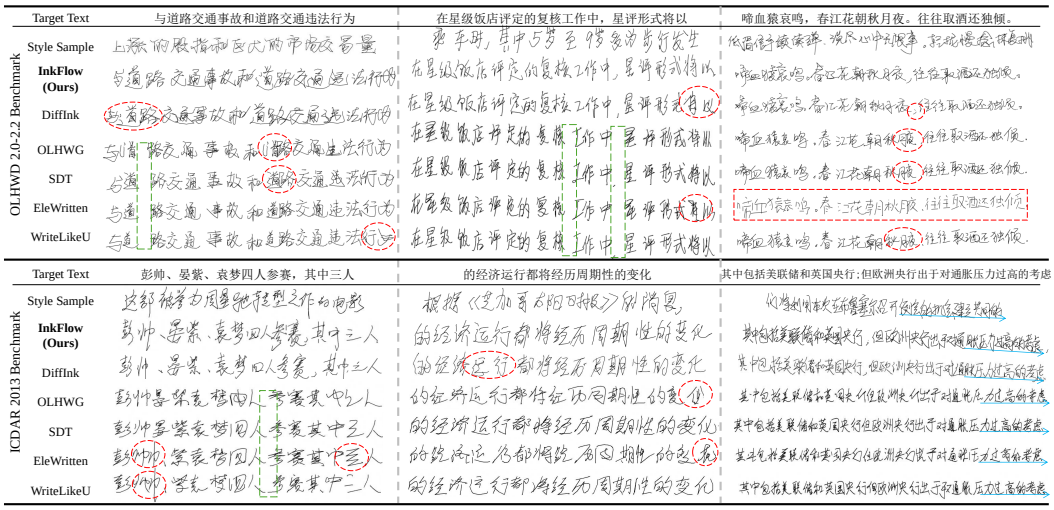


Figure 4: Qualitative comparison of text-line handwriting generation on the Chinese OLHWDB-2.0-2.2 and ICDAR-2013 benchmarks. Red regions indicate character-generation errors, green regions indicate local layout errors, and blue regions highlight global layout trends.

Table 3: Ablation study of perceptual enhancement in the VAE latent space on ICDAR-2013 and OLHWDB-2.0-2.2. The lightweight perceptual regularization improves downstream conditional generation without modifying the diffusion model.

Dataset	VAE Setting	AR% ↑	CR% ↑	FID ↓	HWD ↓
ICDAR-2013	Reconstruction Only	93.35	94.63	4.12	0.32
	+ Perceptual Enhancement	**94.76**	**95.05**	**3.50**	**0.28**
OLHWDB-2.0-2.2	Reconstruction Only	94.80	95.68	2.74	0.33
	+ Perceptual Enhancement	**95.78**	**96.02**	**2.30**	**0.29**

4.4 Ablation Study

Ablation on Task-Aware Latent Space. As shown in Tab. 3, perceptual enhancement consistently improves downstream generation performance on both datasets. By introducing content- and style-aware perceptual enhancement during VAE training, the learned latent space becomes more structured and better aligned with the requirements of subsequent writing flow modeling. Importantly, these gains are obtained without modifying the diffusion model or inference procedure, indicating that a better-structured latent space alone can provide a more effective representation basis for generation. The qualitative results in Fig. 9 further support this observation, showing improved content accuracy in downstream generation.

Ablation on Writing Flow DiT. Tab. 4 evaluates the key designs in Writing Flow DiT through a progressive ablation. The baseline conditions generation only on the reference trajectory latent x_{ref}, without reference text, the Writing-Flow Mask, or Writing-Flow Alignment. Unlike conventional style-transfer methods that typically employ a dedicated style encoder to extract a compact writer-specific representation, our baseline directly uses the VAE latent x_{ref} as the reference condition, without introducing an additional style encoder. We then introduce the reference text T_{ref} to establish the content–trajectory correspondence required for writing-flow modeling, which consistently improves both content accuracy and generation quality. Building on this formulation, the Writing-Flow Mask further regulates information flow between the reference and generation streams and yields substantial gains in AR and CR. Finally, Writing-



Figure 5: **Effect of Writing-Flow Alignment.** t-SNE visualization of five writers in (a) the VAE latent space and (b) the style-aware feature space. Writing-Flow Alignment brings generated samples closer to real handwriting, particularly in the style-aware space.

Table 4: Ablation study of the Writing-Flow Mask and Writing-Flow Alignment in Writing Flow DiT. The Writing-Flow Mask mainly improves content accuracy, while Writing-Flow Alignment further improves generation quality in terms of FID and HWD.

Dataset	Model Variant	AR% ↑	CR% ↑	FID ↓	HWD ↓
ICDAR-2013	Baseline	92.52	93.34	4.51	0.36
	+ Writing-Flow	93.25	93.89	4.20	0.29
	+ Writing-Flow Mask	**94.89**	**95.19**	4.07	0.31
	+ Writing-Flow Alignment	94.76	95.05	**3.50**	**0.28**
OLHWDB-2.0-2.2	Baseline	94.12	94.87	3.13	0.34
	+ Writing-Flow	94.75	95.17	2.75	0.30
	+ Writing-Flow Mask	**95.83**	**96.05**	2.53	0.33
	+ Writing-Flow Alignment	95.78	96.02	**2.30**	**0.29**

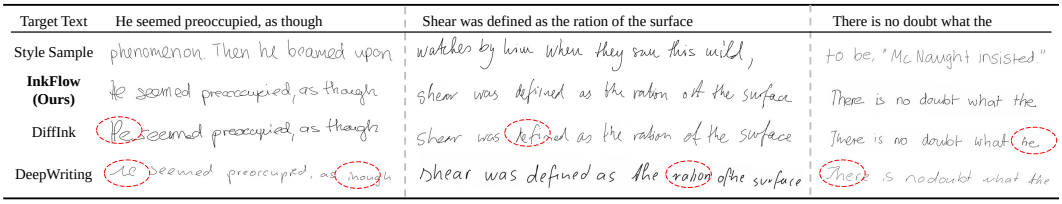


Figure 6: Qualitative comparison of English text-line handwriting generation on the IAM-OnDB benchmark. InkFlow shows improved content fidelity and writer-style consistency compared with DeepWriting and DiffInk.

Flow Alignment improves reference consistency and further reduces FID and HWD. Overall, the results show that writing-flow modeling, structured reference–target interaction, and alignment provide complementary benefits.

The effect of Writing-Flow Alignment is further illustrated in Fig. 5. In the VAE latent space, both variants remain relatively close to real samples, while the difference becomes more evident in the style-aware feature space. With Writing-Flow Alignment, generated samples move closer to the corresponding real-writer clusters, indicating improved alignment with reference writing characteristics without explicitly injecting a style representation.

4.5 Applying to Other Languages

We further evaluate InkFlow on the IAM-OnDB benchmark to examine its applicability to English online handwriting generation. As shown in Tab. 5, InkFlow outperforms DeepWriting (Aksan et al., 2018) and DiffInk (Pan et al., 2026a) across all reported metrics, achieving lower CER and WER as well as improved FID and HWD. The qualitative comparisons in Fig. 6 are consistent with these quantitative gains: InkFlow more faithfully preserves the writing characteristics of the reference samples while maintaining higher content accuracy in the generated text lines. Compared with DiffInk and DeepWriting, the generated handwriting exhibits stronger overall style consistency and a more coherent text-line structure.

Table 5: Quantitative comparison on the IAM-OnDB benchmark.

Method	Venue	CER%↓	WER%↓	FID↓	HWD↓
DeepWriting	CHI'18	8.94	29.68	15.18	0.63
DiffInk	ICLR'26	8.13	25.96	12.48	0.52
InkFlow	**Ours**	**7.98**	**24.48**	**11.41**	**0.51**

5 Conclusion

In this work, we revisit stylized handwriting generation from the perspective of what should be transferred across different contents. Rather than treating writer style as a static representation extracted from observed handwriting, we formulate the transferable factor as the writer-specific content-to-trajectory transformation, termed the **writing flow**. Based on this perspective, we develop **InkFlow** to learn and transfer writing flows from complete handwriting demonstrations in a structured latent space. Through task-aware latent modeling, reference–target joint diffusion, and Writing-Flow Alignment, this formulation is instantiated into a practical generation framework. Extensive experiments on Chinese and English text-line handwriting benchmarks demonstrate strong performance in both content fidelity and writer consistency, supporting writing flow learning as an effective alternative to conventional style-transfer formulations for personalized online handwriting generation.

AI USE STATEMENT

In this work, we used generative AI tools to assist with language editing, as well as environment configuration and data processing during the reproduction of baseline methods. All AI-assisted outputs were reviewed and verified by the authors. We take full responsibility for the final content of this work.

ETHICS STATEMENT

All datasets used in this work are publicly available research datasets and are used in accordance with their original research purposes and licenses. We do not collect or release any additional personally identifiable information. Our method is developed for research on online handwriting generation, and generated samples are intended solely for legitimate research and educational use.

REPRODUCIBILITY STATEMENT

The supplementary material provides detailed descriptions of the model architecture and training configurations. In addition, we will publicly release the source code and model configurations to facilitate future research in this field.

REFERENCES

Emre Aksan, Fabrizio Pece, and Otmar Hilliges. Deepwriting: Making digital ink editable via deep generative modeling. In *Proceedings of the 2018 CHI conference on human factors in computing systems*, pp. 1–14, 2018.

Fan Bao, Shen Nie, Kaiwen Xue, Chongxuan Li, Shi Pu, Yaole Wang, Gang Yue, Yue Cao, Hang Su, and Jun Zhu. One transformer fits all distributions in multi-modal diffusion at scale. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 1692–1717. PMLR, 23–29 Jul 2023.

Haonan Cai, Yuxuan Luo, and Zhouhui Lian. Beyond patches: Global-aware autoregressive model for multimodal few-shot font generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 517–528, June 2026.

Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22560–22570, October 2023.

Mingdeng Cao, Chong Mou, Ziyang Yuan, Xintao Wang, Zhaoyang Zhang, Ying Shan, and Yinqiang Zheng. Consistent human image and video generation with spatially conditioned diffusion. *arXiv preprint arXiv:2412.14531*, 2024.

Jen-Hao Rick Chang, Ashish Shrivastava, Hema Koppula, Xiaoshuai Zhang, and Oncel Tuzel. Style equalization: Unsupervised learning of controllable generative sequence models. In *International Conference on Machine Learning*, pp. 2917–2937. PMLR, 2022.

Weiran Chen, Guiqian Zhu, Ying Li, Yi Ji, and Chunping Liu. Da-font: few-shot font generation via dual-attention hybrid integration. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 6644–6653, 2025.

Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, Jian Zhao, Kai Yu, and Xie Chen. F5-tts: A fairytaler that fakes fluent and faithful speech with flow matching. *arXiv preprint arXiv:2410.06885*, 2024.

Gang Dai, Yifan Zhang, Qingfeng Wang, Qing Du, Zhuliang Yu, Zhuoman Liu, and Shuangping Huang. Disentangling writer and character styles for handwriting generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5977–5986, 2023.

Gang Dai, Yifan Zhang, Quhui Ke, Qiangya Guo, and Shuangping Huang. One-dm: One-shot diffusion mimicker for handwritten text generation. In *European Conference on Computer Vision*, pp. 410–427. Springer, 2024.

Gang Dai, Yifan Zhang, Yutao Qin, Qiangya Guo, Shuangping Huang, and Shuicheng Yan. Beyond isolated words: Diffusion brush for handwritten text-line generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19054–19064, October 2025.

David H Douglas and Thomas K Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: the international journal for geographic information and geovisualization*, 10(2):112–122, 1973.

Bin Fu, Fanghua Yu, Anran Liu, Zixuan Wang, Jie Wen, Junjun He, and Yu Qiao. Generate like experts: Multi-stage font generation by incorporating font transfer process into diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6892–6901, June 2024.

Bin Fu, Zixuan Wang, Kainan Yan, Shitian Zhao, Qi Qin, Jie Wen, Junjun He, and Peng Gao. Fontanimate: High quality few-shot font generation via animating font transfer process. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16015–16025, October 2025.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pp. 369–376, 2006.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 18661–18673. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/d89a66c7c80a29b1bdbab0f2a1a94af8-Paper.pdf.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Hua Li and Zhouhui Lian. Hfh-font: Few-shot chinese font synthesis with higher quality, faster speed, and higher resolution. *ACM Trans. Graph.*, 43(6), November 2024. ISSN 0730-0301. doi: 10.1145/3687994. URL https://doi.org/10.1145/3687994.

Shilong Li, Shusen Tang, and Zhouhui Lian. Chwmaster: mastering chinese handwriting via sliding-window recurrent neural networks. *International Journal on Document Analysis and Recognition (IJDAR)*, 27(3):275–288, 2024.

Cheng-Lin Liu, Fei Yin, Da-Han Wang, and Qiu-Feng Wang. Casia online and offline chinese handwriting databases. In *2011 International Conference on Document Analysis and Recognition*, pp. 37–41. IEEE, 2011.

Yitian Liu and Zhouhui Lian. Qt-font: High-efficiency font synthesis via quadtree-based diffusion models. In *ACM SIGGRAPH 2024 Conference Papers*, SIGGRAPH '24, 2024. URL https://doi.org/10.1145/3641519.3657451.

Yu Liu, Fatimah Binti Khalid, Lei Wang, Youxi Zhang, and Cunrui Wang. Elegantly written: Disentangling writer and character styles for enhancing online chinese handwriting. In *European Conference on Computer Vision*, pp. 409–425. Springer, 2024.

Marcus Liwicki and Horst Bunke. Iam-ondb-an on-line english sentence database acquired from handwritten text on a whiteboard. In *Eighth International Conference on Document Analysis and Recognition (ICDAR'05)*, pp. 956–961. IEEE, 2005.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Canjie Luo, Yuanzhi Zhu, Lianwen Jin, Zhe Li, and Dezhi Peng. Slogan: handwriting style synthesis for arbitrary-length and out-of-vocabulary text. *IEEE transactions on neural networks and learning systems*, 34(11):8503–8515, 2022.

Grace Luo, Trevor Darrell, Oliver Wang, Dan B Goldman, and Aleksander Holynski. Readout guidance: Learning control from diffusion features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8217–8227, June 2024.

Martin Mayr, Marcel Dreier, Florian Kordon, Mathias Seuret, Jochen Zöllner, Fei Wu, Andreas Maier, and Vincent Christlein. Zero-shot paragraph-level handwriting imitation with latent diffusion models. *International Journal of Computer Vision*, 133(10):7054–7075, 2025.

Wei Pan, Anna Zhu, Xinyu Zhou, Brian Kenji Iwana, and Shilin Li. Few shot font generation via transferring similarity guided global style and quantization local style. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19506–19516, 2023.

Wei Pan, Huiguo He, Hiuyi Cheng, Yilin Shi, and Lianwen Jin. Diffink: Glyph- and style-aware latent diffusion transformer for text to online handwriting generation. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=XKOEQFKFdL.

Wei Pan, Xuhan Zheng, Yilin Shi, Huiguo He, Hiuyi Cheng, Dezhi Peng, Minghui Liao, and Lianwen Jin. Diffmath: Symbol-and graph-aware latent diffusion transformer for handwritten mathematical expression generation. *arXiv preprint arXiv:2606.19939*, 2026b.

Song Park, Sanghyuk Chun, Junbum Cha, Bado Lee, and Hyunjung Shim. Few-shot font generation with weakly supervised localized representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(3):1479–1495, 2024. doi: 10.1109/TPAMI.2022.3196675.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

Vittorio Pippi, Fabio Quattrini, Silvia Cascianelli, and Rita Cucchiara. Hwd: A novel evaluation score for styled handwritten text generation. *arXiv preprint arXiv:2310.20316*, 2023.

Vittorio Pippi, Fabio Quattrini, Silvia Cascianelli, Alessio Tonioni, and Rita Cucchiara. Zero-shot styled text image generation, but make it autoregressive. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Min-Si Ren, Yan-Ming Zhang, Qiu-Feng Wang, Fei Yin, and Cheng-Lin Liu. Diff-writer: a diffusion model-based stylized online handwritten chinese character generator. In *International Conference on Neural Information Processing*, pp. 86–100. Springer, 2023.

Minsi Ren, Yan-Ming Zhang, and yi chen. Decoupling layout from glyph in online chinese handwriting generation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=DhHIw9Nbl1.

Douglas Reynolds. Gaussian mixture models. In *Encyclopedia of biometrics*, pp. 827–832. Springer, 2015.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823, 2015.

Yifei Shen, XINYANG JIANG, Yifan Yang, Yezhen Wang, Dongqi Han, and Dongsheng Li. Understanding and improving training-free loss-based diffusion guidance. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=Eu80DGuOcs.

Jinsu Shin, Sungeun Hong, and JinYeong Bak. Cashg: Context-aware stylized online handwriting generation. *arXiv preprint arXiv:2604.02103*, 2026.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Shusen Tang and Zhouhui Lian. Write like you: Synthesizing your cursive online chinese handwriting via metric-based meta learning. In *Computer Graphics Forum*, volume 40(2), pp. 141–151. Wiley Online Library, 2021.

Shusen Tang, Zeqing Xia, Zhouhui Lian, Yingmin Tang, and Jianguo Xiao. Fontrnn: Generating large-scale chinese fonts via recurrent neural network. In *Computer Graphics Forum*, volume 38(7), pp. 567–577. Wiley Online Library, 2019.

Hunyuan Vision Team, Pengyuan Lyu, Xingyu Wan, Gengluo Li, Shangpin Peng, Weinong Wang, Liang Wu, Huawen Shen, Yu Zhou, Canhui Tang, Qi Yang, Qiming Peng, Bin Luo, Hower Yang, Xinsong Zhang, Jinnian Zhang, Houwen Peng, Hongming Yang, Senhao Xie, Longsha Zhou, Ge Pei, Binghong Wu, Rui Yan, Kan Wu, Jieneng Yang, Bochao Wang, Kai Liu, Jianchen Zhu, Jie Jiang, Linus, Han Hu, and Chengquan Zhang. HunyuanOCR Technical Report. *arXiv preprint arXiv:2511.19575*, 2025.

Chi Wang, Min Zhou, Tiezheng Ge, Yuning Jiang, Hujun Bao, and Weiwei Xu. Cf-font: Content fusion for few-shot font generation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1858–1867. IEEE, 2023.

Honglie Wang, Minsi Ren, Yan-Ming Zhang, Fei Yin, and Cheng-Lin Liu. Template-guided cascaded diffusion for stylized handwritten chinese text-line generation. In *Document Analysis and Recognition – ICDAR 2025*. Springer, Cham, 2026a.

Honglie Wang, Yan-Ming Zhang, Wangzi Yao, Fei Yin, and Cheng-Lin Liu. Learning to generate stylized handwritten text via a unified representation of style, content, and noise. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=FBPuLChGNX.

Jiangpeng Wang, Fei Gao, and Nannan Wang. Hierarchical style aggregation for versatile chinese handwriting generation. In *Computer Vision – ECCV 2026*, pp. 104–122, 2026c.

Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *European conference on computer vision*, pp. 499–515. Springer, 2016.

Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, and Saining Xie. Convnext v2: Co-designing and scaling convnets with masked autoencoders. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16133–16142, 2023.

Serin Yang, Hyunmin Hwang, and Jong Chul Ye. Zero-shot contrastive loss for text-guided diffusion image style transfer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22873–22882, October 2023.

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. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 6603–6611, 2024.

Fei Yin, Qiu-Feng Wang, Xu-Yao Zhang, and Cheng-Lin Liu. Icdar 2013 chinese handwriting recognition competition. In *2013 12th international conference on document analysis and recognition*, pp. 1464–1470. IEEE, 2013.

Xu-Yao Zhang, Fei Yin, Yan-Ming Zhang, Cheng-Lin Liu, and Yoshua Bengio. Drawing and recognizing chinese characters with recurrent neural network. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):849–862, 2017.

Anna Zhu, Wei Pan, Guan Li, Hongyi Cai, and Brian Iwana Kenji. Hq-font: Few-shot font generation via transferring hierarchical quantization styles. *International Journal on Document Analysis and Recognition*, 2026.

APPENDIX

A IMPLEMENTATION DETAILS

A.1 DATA REPRESENTATION

As illustrated in Fig. 7, each online handwriting sample is represented as an ordered sequence $X \in \mathbb{R}^{N\times 4}$, where each point is represented by (x, y, s_1, s_2). Here, (x, y) denotes the spatial coordinate of the pen tip, while (s_1, s_2) encodes the binary pen state. Connecting consecutive points according to their pen states recovers the original handwriting trajectory.

We first normalize the (x, y) coordinates of all trajectory points following prior work (Zhang et al., 2017), reducing scale variations caused by differences in device resolution and handwriting size. We then apply the Ramer–Douglas–Peucker (RDP) algorithm (Douglas & Peucker, 1973) to each stroke independently to remove redundant trajectory points while preserving the overall handwriting structure. For CASIA-OLHWDB, the simplification coefficient is empirically set to 0.4. This preprocessing substantially shortens the trajectory sequences and reduces the computational cost of subsequent sequence modeling.

For mini-batch training, handwriting sequences are padded to the maximum sequence length within each batch. The coordinate dimensions of padded positions are filled with zeros, and a corresponding sequence mask is used to exclude padded regions from model computation and loss calculation.

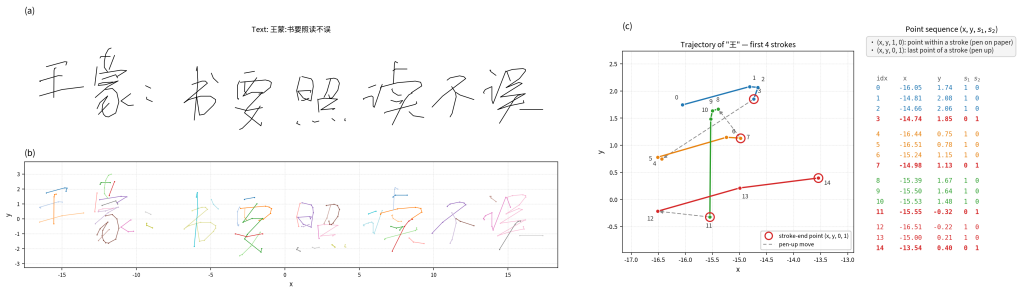


Figure 7: **Online handwriting data representation.** (a) Rendered text-line image of an online handwriting sample. (b) The corresponding handwriting trajectory represented in the 2D coordinate space, where different colors indicate individual strokes. (c) Detailed representation of the first character, illustrating its ordered point sequence and the associated pen-state information.

A.2 DETAILS OF TASK-AWARE VAE

Reconstruction and KL Regularization. As the latent modeling module of InkFlow, the task-aware VAE is designed to provide compact handwriting representations for subsequent writing flow learning. Given an online handwriting trajectory $X \in \mathbb{R}^{N\times 4}$, the VAE adopts a 1D convolutional encoder–decoder architecture with residual connections. The encoder progressively downsamples the input sequence into representations of size $N/2 \times 128$, $N/4 \times 256$, and finally $N/8 \times 384$, resulting in an $8\times$ temporal compression ratio. The decoder mirrors this structure and progressively restores the temporal resolution. A Transformer-based trajectory decoder with 3 layers, hidden size 256, and 4 attention heads is further employed for sequence modeling.

The final decoder output has dimension $N \times 122$, where 120 dimensions parameterize a 20-component Gaussian Mixture Model (GMM) for continuous (x, y) coordinate prediction and the remaining 2 dimensions are used for pen-state classification. For each valid timestep t, let $\mathbf{p}_t = (x_t, y_t)$ denote the ground-truth coordinate. The GMM reconstruction loss is defined as

$$\mathcal{L}_{\text{gmm}} = -\frac{1}{T}\sum_{t=1}^{T}\log\left(\sum_{m=1}^{M}\pi_{m,t}\mathcal{N}\left(\mathbf{p}_t \mid \boldsymbol{\mu}_{m,t}, \boldsymbol{\Sigma}_{m,t}\right)\right), \tag{11}$$

where $M = 20$, and $\pi_{m,t}$, $\boldsymbol{\mu}_{m,t}$, and $\boldsymbol{\Sigma}_{m,t}$ denote the mixture weight, mean, and covariance of the m-th Gaussian component.

Pen-state prediction is formulated as a 2-way classification problem and optimized using focal loss:

$$\mathcal{L}_{\text{pen}} = -\frac{1}{T}\sum_{t=1}^{T}\sum_{k=1}^{2}\alpha_k(1-p_{t,k}^{\text{pen}})^{\gamma}y_{t,k}^{\text{pen}}\log p_{t,k}^{\text{pen}}, \tag{12}$$

where $y_{t,k}^{\text{pen}}$ and $p_{t,k}^{\text{pen}}$ denote the ground-truth label and predicted probability of pen-state class k, respectively.

Following the standard VAE formulation (Kingma & Welling, 2013), a KL divergence term is additionally used to regularize the posterior distribution:

$$\mathcal{L}_{\text{kl}} = D_{\text{KL}}\left(q_\phi(x|X) \parallel \mathcal{N}(0,I)\right). \tag{13}$$

Content- and Style-Aware Perceptual Enhancement. Beyond trajectory reconstruction, we introduce lightweight perceptual supervision to encourage the latent space to preserve both textual structure and writer-specific handwriting characteristics.

For content awareness, a Transformer-based recognition head $\mathcal{H}_{\text{ctc}}$ predicts character sequences from the latent representation and is optimized with CTC loss (Graves et al., 2006):

$$\mathcal{L}_{\text{ctc}} = \text{CTC}\left(\mathcal{H}_{\text{ctc}}(x), \mathbf{y}^{\text{text}}\right). \tag{14}$$

For style awareness, a style-aware head $\mathcal{H}_{\text{sty}}$ maps the latent sequence to a global writer representation $\mathbf{s} = \mathcal{H}_{\text{sty}}(x)$. We jointly employ writer classification, center loss (Wen et al., 2016), and triplet loss (Schroff et al., 2015):

$$\mathcal{L}_{\text{sty}} = \mathcal{L}_{\text{cls}} + 0.01\mathcal{L}_{\text{cen}} + \mathcal{L}_{\text{tri}}. \tag{15}$$

The center loss encourages intra-writer compactness,

$$\mathcal{L}_{\text{cen}} = \frac{1}{B}\sum_{i=1}^{B}\|\mathbf{s}_i - \mathbf{c}_{y_i}\|_2^2, \tag{16}$$

while the triplet loss improves inter-writer separability:

$$\mathcal{L}_{\text{tri}} = \frac{1}{|\mathcal{T}|}\sum_{(a,p,n)\in\mathcal{T}}\left[\|\mathbf{s}_a - \mathbf{s}_p\|_2 - \|\mathbf{s}_a - \mathbf{s}_n\|_2 + m\right]_+, \tag{17}$$

where a, p, and n denote anchor, positive, and negative samples, respectively.

The overall VAE objective is

$$\mathcal{L}_{\text{VAE}} = \mathcal{L}_{\text{gmm}} + \mathcal{L}_{\text{pen}} + 10^{-6}\mathcal{L}_{\text{kl}} + 0.1\,\mathcal{L}_{\text{ctc}} + 0.1\,\mathcal{L}_{\text{sty}}. \tag{18}$$

After training, the VAE and the two perceptual heads are frozen for subsequent writing flow learning. The style-aware feature space learned by $\mathcal{H}_{\text{sty}}$ is further reused for Writing-Flow Alignment during diffusion training, providing reference-aware supervision without introducing an additional perceptual encoder.

A.3 Details of Writing Flow DiT

Text Encoder. The reference and target text sequences are first mapped to learnable character embeddings. Since the text sequence length generally differs from the corresponding handwriting latent length, the character sequence is padded to match the latent resolution using a shared filler token. The resulting embeddings are processed by a lightweight text encoder composed of ConvNeXt-V2 (Woo et al., 2023) blocks. Each block performs local contextual modeling through depthwise 1D convolution, followed by normalization and feed-forward transformation. Positional information is additionally injected before the ConvNeXt-V2 blocks. The encoded reference and target text features are then used as textual conditions for the corresponding handwriting streams.

Input Projection and Flow Construction. Writing Flow DiT adopts stream-specific input projections for the reference and generation branches. Specifically, the reference handwriting latent and reference text feature are independently projected into the DiT hidden space using two linear layers, while another pair of projection layers is used for the target noisy latent and target text feature. All projected features have a hidden dimension of 896.

Within the reference stream, the projected reference latent and reference text feature are added and normalized by a dedicated LayerNorm. The same operation is performed independently for the generation stream. Learnable role embeddings are then added to distinguish reference tokens from generation tokens. To further capture local temporal dependencies, a convolutional positional embedding is applied separately to each stream.

The resulting reference and generation features are finally concatenated along the sequence dimension and jointly processed by the Transformer backbone. This stream-specific projection design allows the model to preserve the distinct roles of the reference and generation inputs before their interaction in the shared DiT feature space.

DiT Backbone. Writing Flow DiT consists of 18 Transformer blocks with a hidden dimension of 896. Each block contains multi-head self-attention and a feed-forward network, while diffusion timestep information is injected through timestep-conditioned adaptive normalization.

Rotary positional embeddings (RoPE) (Su et al., 2024) are applied in the self-attention layers. The reference and generation streams independently start their positional indices from zero, preserving their respective temporal structures after concatenation. The asymmetric Writing-Flow Mask introduced in the main paper is applied throughout all Transformer blocks to regulate directional information flow between the two streams.

After the Transformer backbone, only the output tokens corresponding to the generation stream are retained. These features are passed through a timestep-conditioned final normalization layer and a linear prediction head, which maps the 896-dimensional hidden representation back to the 384-dimensional VAE latent space to obtain the predicted clean latent $\hat{x}_0$.

Classifier-Free Guidance and Inference. During inference, we construct three prediction branches: the full conditional prediction, the prediction without the reference condition, and the unconditional prediction without either the reference or target-text condition. The final prediction is obtained by independently combining content and reference guidance:

$$\hat{x}_0 = \hat{x}_0^{\text{uncond}} + w_c \left(\hat{x}_0^{\text{ref-free}} - \hat{x}_0^{\text{uncond}}\right) + w_r \left(\hat{x}_0^{\text{full}} - \hat{x}_0^{\text{ref-free}}\right), \tag{19}$$

where w_c and w_r control the target-content and reference-writing-flow guidance strengths, respectively. We set $w_c = 2.0$ and $w_r = 1.5$ in all experiments. DDIM (Song et al., 2020) sampling is used for generation with 20 inference steps and $\eta = 0$. Starting from Gaussian noise, the target latent is progressively denoised and finally decoded by the frozen VAE decoder into an online handwriting trajectory.

A.4 Evaluation Metrics

To evaluate textual correctness, we employ a self-deployed HunyuanOCR (Team et al., 2025) model as the recognition evaluator. On real test samples, it achieves 96.92% AR and 97.16% CR on OLHWDB-2.0-2.2 (Liu et al., 2011), 94.18% AR and 94.64% CR on ICDAR-2013 Yin et al. (2013), and 2.20% CER and 13.55% WER on IAM-OnDB. For Chinese handwriting, we report Accurate Rate (AR) and Correct Rate (CR) following (Yin et al., 2013), where N_t is the number of ground-truth characters and D_e, S_e, and I_e denote deletion, substitution, and insertion errors:

$$\text{AR} = \frac{N_t - D_e - S_e - I_e}{N_t} \times 100\%, \qquad \text{CR} = \frac{N_t - D_e - S_e}{N_t} \times 100\%. \tag{20}$$

For English handwriting on IAM-OnDB, we report Character Error Rate (CER) and Word Error Rate (WER), both computed based on edit distance.

For generation quality, Fréchet Inception Distance (Heusel et al., 2017) (FID) is computed on complete rendered text-line images. HWD (Pippi et al., 2023) is evaluated using its official implementation, without resizing either real or generated handwriting samples.

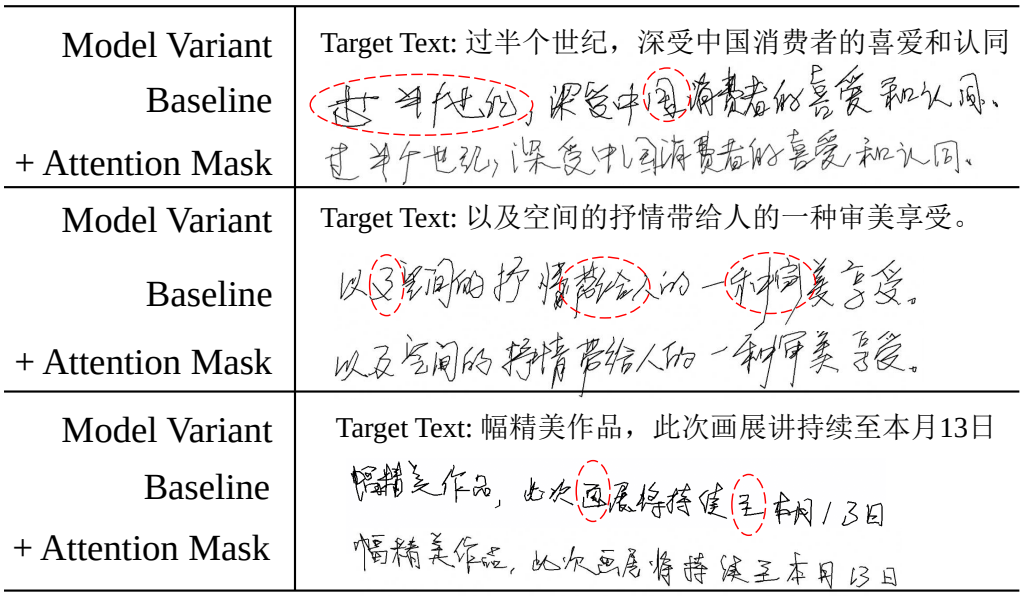


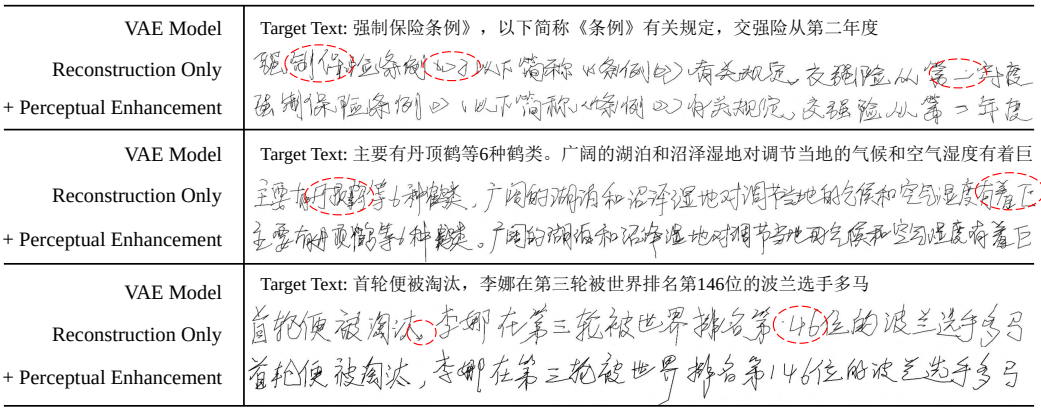
Figure 8: **Visual ablation of the Writing-Flow Mask in Writing Flow DiT.** We compare text-line generation results with and without the proposed Writing-Flow Mask. Red regions indicate content generation errors. The Writing-Flow Mask improves content fidelity by better regulating reference–target interaction during writing-flow transfer.

B Additional Experimental Results

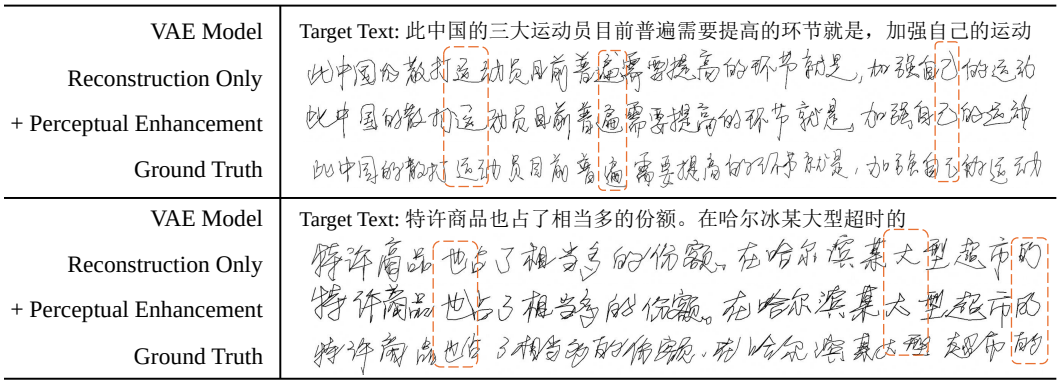
B.1 Visual Ablation Studies

We provide additional qualitative ablations to complement the quantitative analysis in the main paper. We first visualize the effect of the asymmetric Writing-Flow Mask in Writing Flow DiT in Fig. 8. Compared with the variant without the proposed mask, introducing structured reference–target interaction reduces content generation errors, as indicated by the red regions. These observations are consistent with the quantitative improvements in AR and CR reported in the main paper, suggesting that restricting the noisy generation stream from perturbing the reference representation provides a more stable condition for writing-flow transfer.

Fig. 9 further illustrates the effect of perceptual enhancement in the task-aware VAE on downstream handwriting generation. As shown in Fig. 9a, content-aware perceptual enhancement helps the latent representation preserve textual structure more effectively. Without this enhancement, the downstream generation model is more likely to produce character omissions, repetitions, or incorrect character quantities, as highlighted by the red regions. This effect is particularly noticeable for long Chinese text lines, where maintaining accurate content structure becomes more challenging. Fig. 9b further shows that style-aware perceptual enhancement facilitates more consistent preservation of writer-specific handwriting characteristics during downstream generation, as evidenced by the highlighted regions.



(a)



(b)

Figure 9: **Effect of VAE perceptual enhancement on downstream generation.** (a) Content-aware perceptual enhancement improves downstream content generation accuracy. Red regions indicate character errors produced when using a VAE trained only with reconstruction loss, showing that content-aware features better preserve character quantity and sequence structure. (b) Style-aware perceptual enhancement improves writer-style consistency in downstream generation. Yellow regions highlight local handwriting patterns that provide visual evidence of improved consistency with the reference style.

B.2 Additional Qualitative Comparisons and Generation Results

We further provide more qualitative comparisons with representative baselines for both Chinese and English handwriting generation. Fig. 10 presents additional Chinese text-line results, where the red regions highlight content generation errors and the green regions indicate unnatural local layouts. Compared with baseline methods, InkFlow generally produces more accurate textual content while maintaining more coherent character arrangement and text-line structure. Additional English comparisons on IAM-OnDB are shown in Fig. 11. The red regions highlight content that baseline methods fail to generate correctly, while InkFlow produces more reliable textual outputs and better preserves the writing characteristics demonstrated by the reference samples.

Finally, we present additional generation results of InkFlow itself to further illustrate its ability to transfer writing characteristics across different contents. Fig. 12 shows Chinese generation examples under multiple writing styles, while Fig. 13 presents the corresponding results for English handwriting. For each writing style, the first row provides the reference handwriting and the remaining rows contain generated text lines with different contents. Across both languages, InkFlow maintains recognizable writer-specific characteristics across multiple generated contents, demonstrating the applicability of writing-flow learning to diverse text-line generation scenarios.



Figure 10: **More Chinese text-line generation results on OLHWDB-2.0-2.2.** We present additional qualitative comparisons of InkFlow with baseline methods. Red regions indicate content generation errors, while green regions highlight unnatural local layouts within the generated text lines.

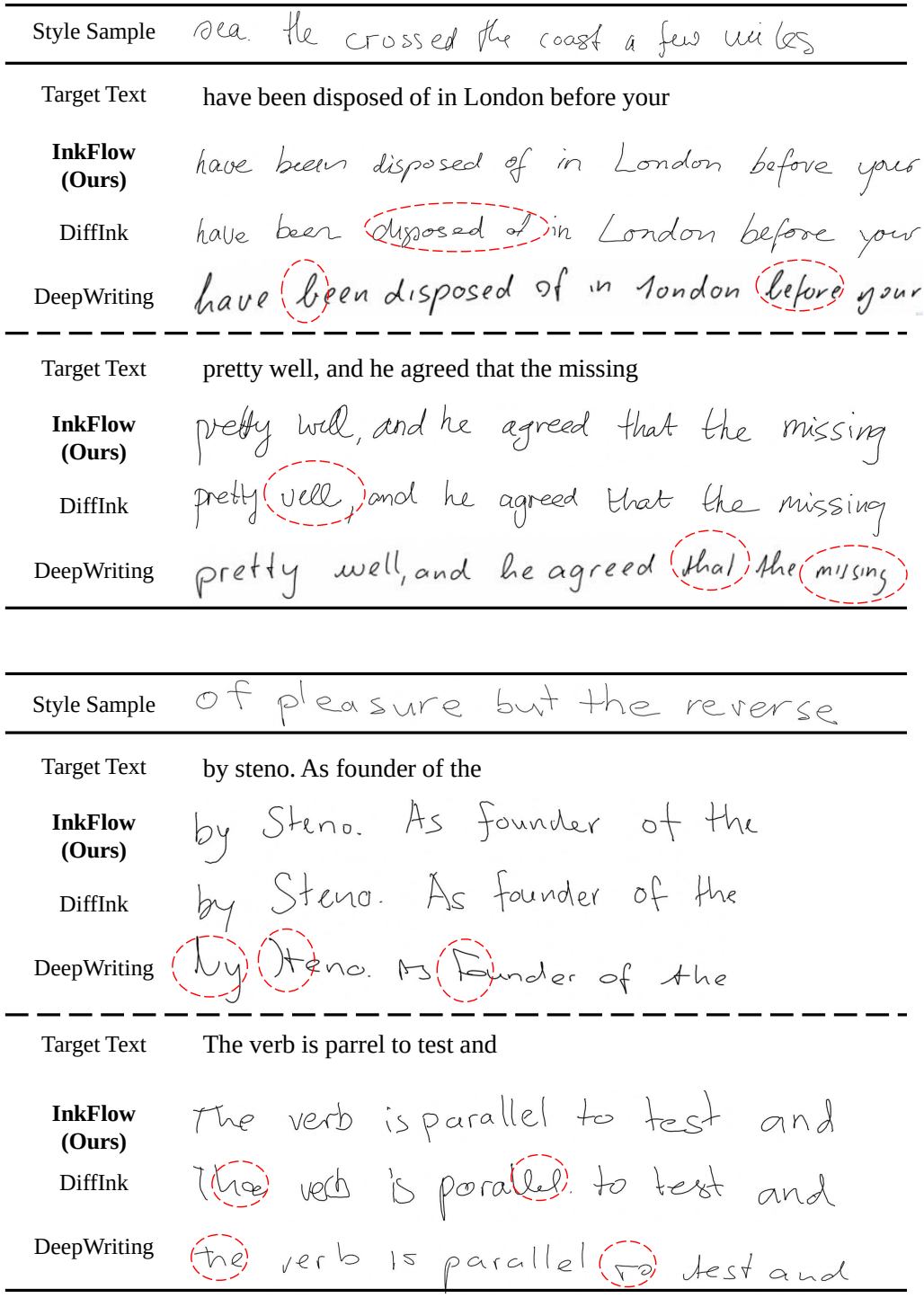


Figure 11: **More qualitative comparisons on the IAM-OnDB benchmark.** We present additional English text-line generation results of InkFlow and baseline methods. Red regions highlight textual content that baseline methods fail to generate correctly.

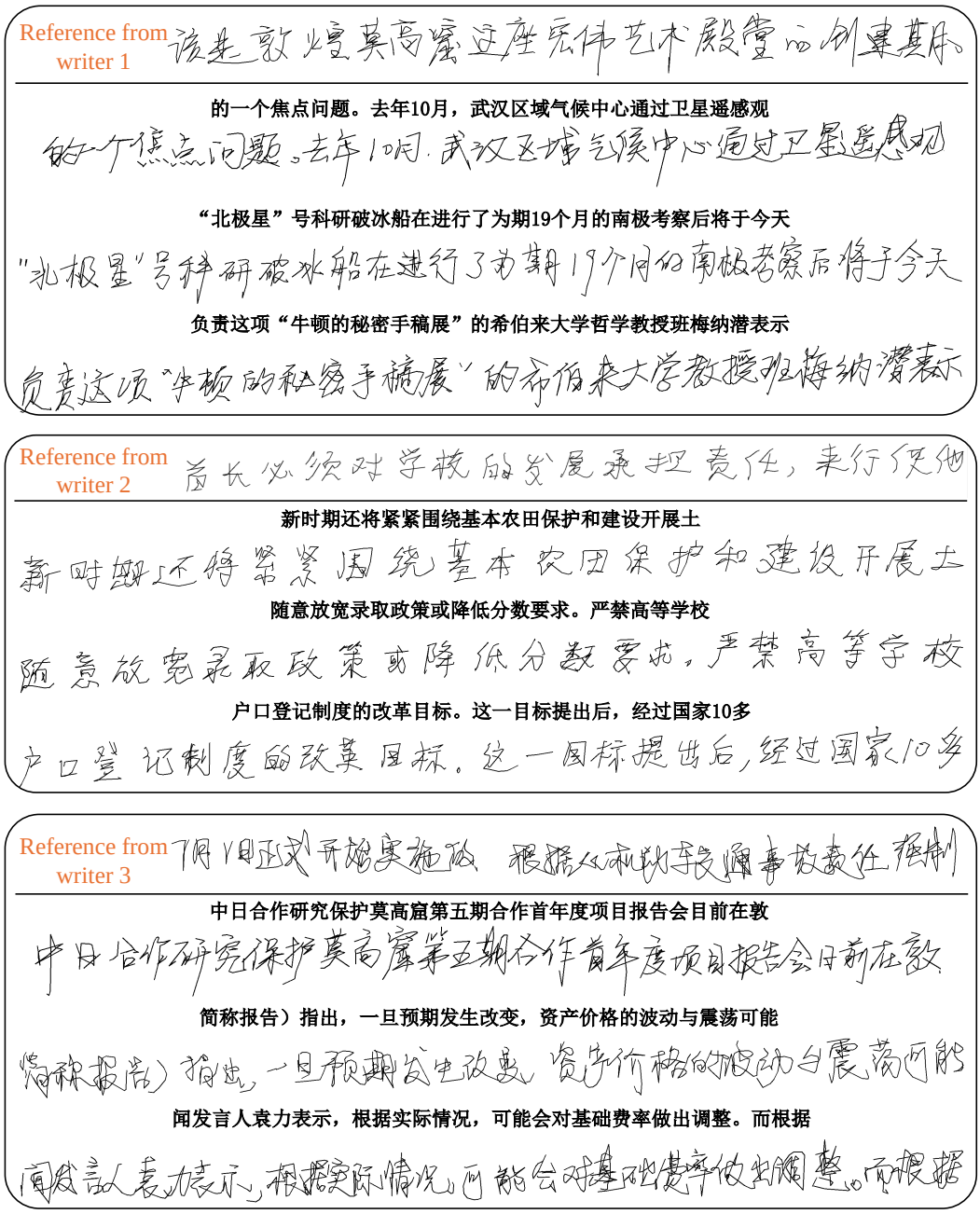


Figure 12: **More generation results by InkFlow (Ours).** For each writing style, the first row shows the reference handwriting samples, while the remaining rows present text lines generated by InkFlow under the same style condition.

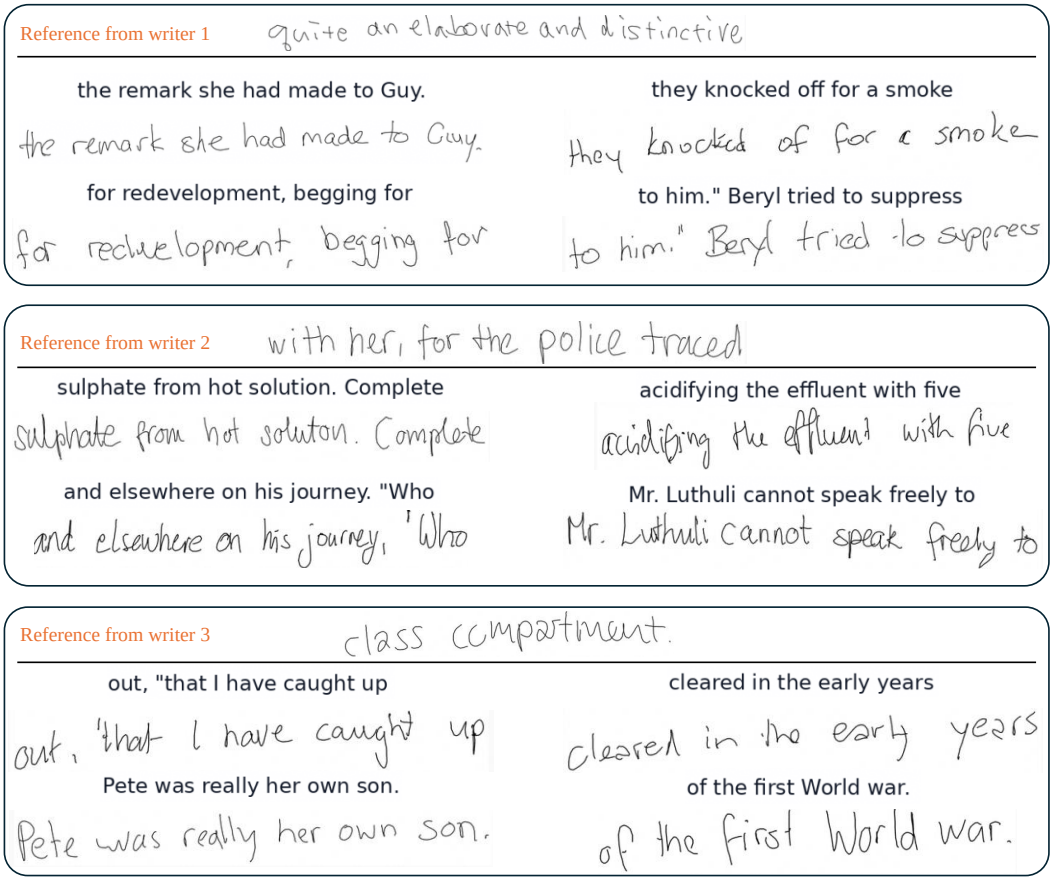


Figure 13: **More English generation results by InkFlow (Ours)** For each writing style, the first row shows the reference handwriting samples, while the remaining rows present English text lines generated by InkFlow under the same writing style.