InkFormer: From Characters to Signatures via Progressive Style-Content Decoupled Online Handwriting Generation

Anonymous authors
Paper under double-blind review

Abstract

Online handwritten signatures provide dynamic signing trajectories and serve as an important biometric trait for identity authentication. However, online signature verification remains constrained by the limited scale and diversity of training data, particularly when forgery samples are unavailable or only a single genuine signature is accessible. Although generative modeling can alleviate data scarcity, existing online handwriting generators remain difficult to adapt to signature synthesis due to limited style controllability and insufficient multi-character structure modeling. Since signatures can be regarded as writer-specific online handwriting trajectories, we rethink signature synthesis as controllable online handwriting generation and propose **InkFormer**, an auto-regressive framework for style-controllable online handwriting generation. InkFormer includes three key designs: self-constructed online standard characters for modality-consistent content guidance, feature-level content–style residuals for explicit writer-style modeling, and a style–content-aware learnable starting token for auto-regressive decoding. With content–style cross-attention during generation, InkFormer produces high-fidelity handwriting trajectories with accurate structures and consistent styles. Building upon InkFormer, we develop a progressive signature generation framework that transfers model capability from character-level to word-level generation, and finally to online signature synthesis. The model first learns character structures and writer styles, then strengthens multi-character and inter-character modeling through word-level fine-tuning. Finally, real signatures are used as style references and signature contents as prompts to generate online signature trajectories for verification training. Experiments show that InkFormer achieves superior online handwriting generation quality over existing methods. Moreover, as a data augmentation engine for signature verification, the generated signatures significantly improve verification performance under forgery-free and one-shot genuine low-resource scenarios. The source code will be publicly available.

1 Introduction

Signatures are widely used for identity authentication in financial, legal, and administrative scenarios, making signature verification a critical biometric technology. Compared with offline signatures, online signatures capture dynamic writing trajectories (e.g., stroke order and velocity), providing richer discriminative cues for verification. Existing approaches typically learn high-dimensional representations of temporal signatures using neural networks, and combine them with similarity metrics or auxiliary classifiers (Jiang et al., 2022; Tolosana et al., 2021b; Yu & Shi, 2023; Shi et al., 2025; Hai et al., 2025). However, real-world signature verification suffers from severe data scarcity, as genuine signatures are hard to collect at scale due to privacy constraints, while forgery samples are even more costly to obtain (Bhavani & Bharathi, 2024; Kaur & Kumar, 2023; Liu et al., 2023). Consequently, practical systems are often trained under forgery-free or one-shot genuine scenarios, making it challenging to learn generalizable signature representations. As illustrated in fig. 1, most existing signature synthesis methods rely on the re-parameterization mechanism (Ferrer et al., 2019; Diaz et al., 2016; Lai et al., 2022; Tolosana et al., 2021a) or temporal predictors (Pandey et al., 2024; 2025). Such approaches remain confined to locally constrained perturbations of existing signatures, resulting in limited diversity and weakened generalization of verification models in real-world scenarios. In contrast,

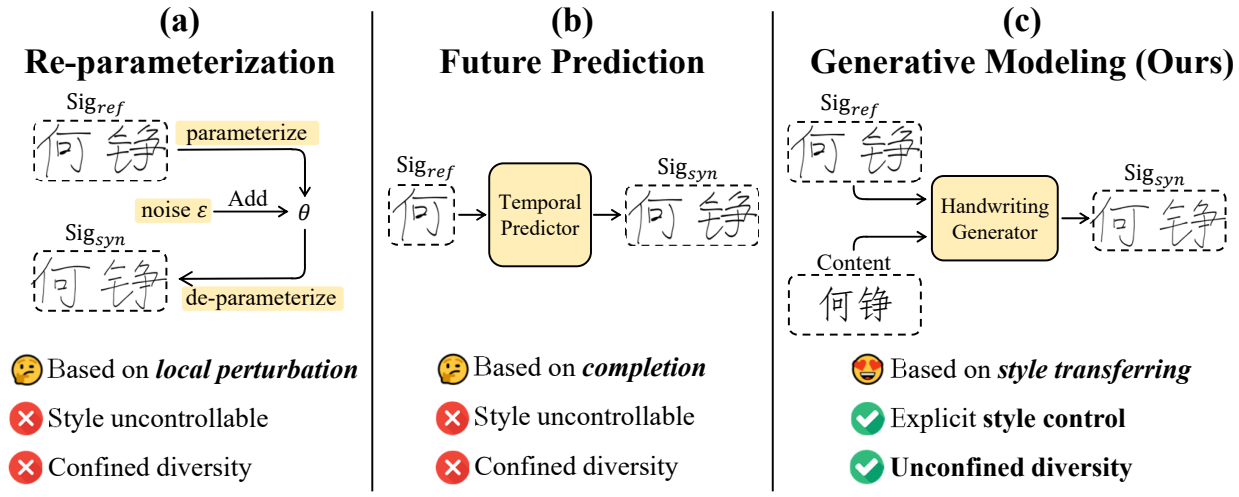


Figure 1: Comparison of different signature synthesis methods. Both (a) re-parameterization-based and (b) temporal future prediction method lack style control and are confined in diversity, whereas the proposed (c) generative modeling method enables explicit style control and diverse signature generation.

generative modeling provides a more principled alternative by learning the underlying distribution of handwriting styles, where real signatures can serve as style references rather than direct transformation anchors. This paradigm offers greater flexibility and fidelity for signature synthesis.

In principle, a well-designed online handwriting generator with strong style and content controllability could be naturally extended to signature generation. However, existing online handwriting generation methods still face several fundamental limitations (Chen et al., 2025; Diaz et al., 2025; Wang et al., 2025b). First, most prior works are confined to character-level generation and lack the capability to model multi-character sequential structures, making them ill-suited for signature generation. Second, style modeling remains insufficiently explored: many approaches do not explicitly disentangle style from content, resulting in weak style consistency. Third, unlike offline handwriting (Dai et al., 2024; 2025; Nikolaidou et al., 2024; Pippi et al., 2025; Wang et al., 2026; 2025a) or font generation (Liu & Lian, 2024; Pan et al., 2023; Fu et al., 2025), online handwriting generation often lacks same-modality content prompts, creating a modality gap between conditional inputs and generated outputs. These limitations not only restrict the expressive power of current online handwriting generators, but also prevent their effective transfer to signature generation.

To address the aforementioned challenges, we propose a signature generation framework built upon a controllable online handwriting generation model, providing a more interpretable and reliable solution for improving online signature verification under data-scarce scenarios. Specifically, we first design **InkFormer-Char**, which introduces a feature-differencing–based style–content decoupling encoder together with an online content prompting mechanism, enabling precise content control while preserving style consistency. Next, we fine-tune the model on word-level data to obtain **InkFormer-Word**, equipping it with multi-character sequential modeling capability, thus making it suitable for signature generation. Finally, we leverage the generation capability of InkFormer-Word to synthesize signature samples and design a signature-oriented post-processing strategy to further enhance the quality of the generated signatures.

Our main contributions can be summarized as follows: 1) We propose InkFormer, a controllable online handwriting generation model that achieves state-of-the-art performance on the ICDAR-2013 benchmark, enabling high-fidelity and style-consistent online handwritten character generation. 2) We improve handwriting style and content modeling through two key designs: residual style decoupling for extracting character-independent writer-level style cues, and modality-consistent content prompting based on collected online standard character trajectories. 3)We extend InkFormer to word-level and signature-level generation, and demonstrate its effectiveness as a generative engine for online signature verification, significantly improving verification performance under forgery-free and one-shot genuine settings.

2 Related Work

Online Handwriting Generation. Online handwriting generation has been predominantly modeled using autoregressive architectures to capture causal temporal dependencies. Early approaches (Graves, 2013; Aksan et al., 2018; Zhang et al., 2017; Ha & Eck, 2017; Das et al., 2020) relied on shallow RNNs without considering style control. Later studies explored various style encoding designs: DeepImitator (Zhao et al., 2020) designed a meta-style matrix to achieve adaptive style generalization; WLU (Tang & Lian, 2021) introduced an attention-based style block in the decoder to enable adaptive style integration; SDT (Dai et al., 2023) employed a dual-head style encoder to separately supervise writer-related and character-related styles, achieving finer-grained style modeling; ElegantlyWritten (Liu et al., 2024) and EHW-Font (Wang et al., 2025c) constructed quantized style encoding to enhance handwriting aesthetics. Following ChiroDiff (Das et al., 2023), diffusion-based models have been increasingly explored for online handwriting generation (Ren et al., 2023; 2025; Pan et al., 2026). Beyond character-level generation, some works have investigated line-level handwriting generation. OLHWG (Ren et al., 2025) implemented a divide-and-conquer strategy by constructing separate pipelines for character generation and layout generation. CHWmaster (Li et al., 2024), within the autoregressive paradigm, controlled the prefix length to generate at the line level. More recently, DiffInk (Pan et al., 2026) proposed a non-autoregressive end-to-end line-level generation paradigm, which significantly improves generation efficiency while maintaining high output quality. Overall, existing methods achieve satisfactory fidelity in content generation, yet the modeling of handwriting style still requires further exploration. In addition, due to the absence of online standard characters, most works rely on dictionaries or offline standard characters for content prompts, introducing a gap in online generation tasks. These challenges also limit the extension of current methods to other domains, such as signature generation tasks that demand extremely high fidelity in both style and character structure.

Signature Synthesis. Existing online signature synthesis methods largely rely on reparameterization mechanisms: Multiple methods (Lai et al., 2022; Ferrer et al., 2019; Diaz et al., 2016) utilize Sigma-LogNormal model (Plamondon, 1995) and generate genuine and forgery samples by perturbing the $\Sigma\Lambda$ parameter space; DeepWriterSyn (Tolosana et al., 2021a) designs a stroke-level VAE framework, extending signature diversity via posterior sampling. In addition, several approaches (Pandey et al., 2024; 2025) treat signature synthesis as a temporal future prediction problem, yet the impact of its synthetic data on verification performance remains largely unexplored. Notably, most existing methods focus on augmentation rather than generative modeling, resulting in limited transformation diversity and reduced generalization in real-world scenarios. Furthermore, these studies have primarily advanced Latin-alphabet signature datasets, while their effectiveness and performance on Chinese signatures remain underexplored.

In light of these observations, this paper starts from online handwritten Chinese character generation and develops **InkFormer**, which features enhanced style modeling and content prompting. Furthermore, we propose a progressive transfer strategy from character generation to signature generation, aiming to bridge the gap between the two tasks while providing a stable and high-quality solution for signature data augmentation.

3 Method

3.1 Overall pipeline

As illustrated in fig. 2, our framework synthesizes Chinese signatures through a three-stage progressive transfer process, extending an online handwriting generator from character-level modeling to signature generation:

(1) Character-level Modeling. We first train a character generation model, termed **InkFormer-Char**, on large-scale online handwritten Chinese character data. Within an autoregressive framework, the model is trained to jointly model writing style and content. This stage allows the model to acquire strong character-structure understanding and writer-style modeling capability from large-scale handwriting data.

(2) Word-level Adaptation. Building upon the style–content modeling capability learned at character level, we further fine-tune the model using online handwritten Chinese word data, resulting in **InkFormer-Word**. This stage enables the model to extend its generation ability from isolated characters to multi-character sequences, and to model longer temporal dependencies in continuous handwriting.

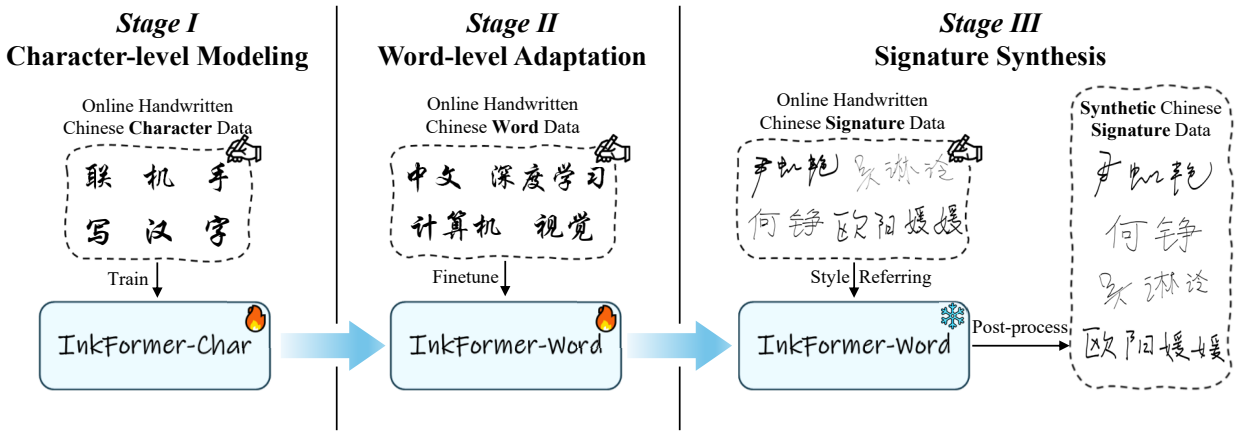


Figure 2: **Overview of the proposed framework**. The framework consists of three progressive stages: (1) Character-level Modeling, (2) Word-level Adaptation, and (3) Signature Synthesis. The framework incrementally transfers InkFormer's capability from character generation to signature generation.

(3) Signature Synthesis. Finally, we use real signatures as style references and signature contents as prompts to synthesize style-consistent online signature trajectories, which are used to augment verification training data and improve signature verification under data-scarce settings.

3.2 InkFormer-Char Modeling

To model the sequential nature of online handwriting, we construct InkFormer-Char under an auto-regressive generation paradigm. As illustrated in fig. 4, the model consists of four main components: style encoding, content encoding, SOS generation, and conditional trajectory generation.

Data Format. An online handwritten character Char can be represented as a trajectory sequence of length L, i.e. $\text{Char} = \{p_1, p_2, \cdots, p_L\}$, where $p_i = (x_i, y_i, pen_i)$. Each trajectory point $p_i \in \mathbb{R}^5$ consists of a two-dimensional coordinate (x_i, y_i) and a three-dimensional one-hot vector pen_i indicating the pen states: *Pen Down*, *Pen Up*, *End of Writing*.

Problem Formulation. In the task of handwritten text generation (HTG), the model takes K characters from a target writer as style reference, $R_{\text{sty}} = \{R^1_{\text{sty}}, R^2_{\text{sty}}, \cdots, R^K_{\text{sty}}\}$, and uses a content reference character R_{con} as content guidance. The goal is to generate online handwriting trajectories that are stylistically consistent with the reference writer while matching the specified character content.

Trajectory-Based Content Encoding. Existing handwriting generation methods commonly use character embeddings or offline character images as content references (Pan et al., 2023; Dai et al., 2024; Ren et al., 2023; 2025). However, character embeddings are limited to a predefined vocabulary and struggle to generalize to unseen characters, while offline images introduce a modality gap with online trajectory generation. To provide content guidance that is both structurally explicit and modality-consistent, we collect and process over 9,000 online standard Chinese character trajectories to construct a standard character dataset. These trajectories provide stable stroke structures and keep the content prompts in the same modality as the generated handwriting. Several character examples are shown in the fig. 3. These online standard characters provide stable stroke structures and keep the content prompts in the same modality as the generated trajectories. We employ a 1D MobileNetV3 (Howard et al., 2019) as the content encoder to extract the content feature f_{con} from R_{con}.

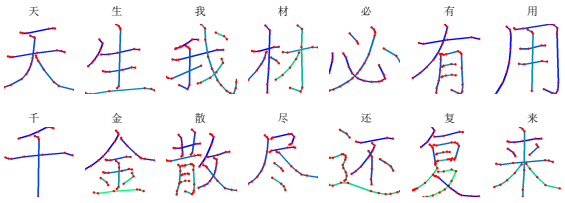


Figure 3: Examples from the constructed online standard character dataset.

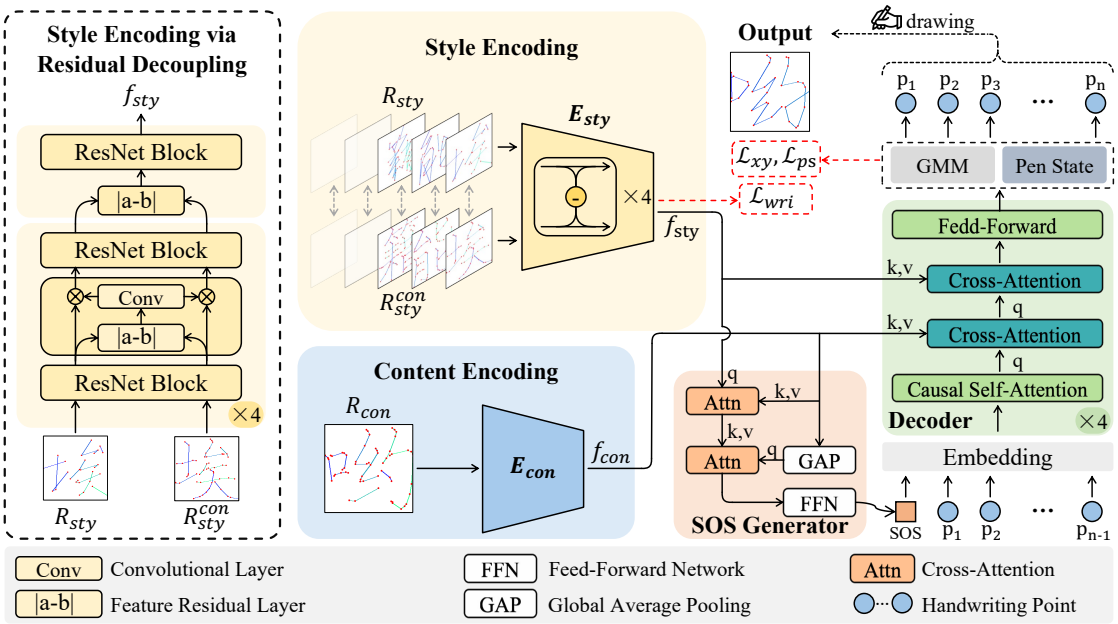


Figure 4: **Overview of InkFormer.** InkFormer extracts residual writer-style features from handwritten references and encodes online standard-content characters as content guidance. The style and content features are fused to generate an adaptive SOS token, which initializes a conditional trajectory generator for auto-regressive prediction of coordinates and pen states.

Style Encoding via Residual Decoupling. For handwritten character and signature generation, effective style modeling should go beyond reproducing reference appearances and capture character-independent writer-level traits. Existing methods (Dai et al., 2023; Pan et al., 2023; Dai et al., 2024) typically encode limited style references for generation and rely on style-related losses to encourage writer consistency, but they often overlook the influence of character structures on style extraction. Since different characters may exhibit substantial structural variations, the learned style representation can be disturbed by content-related structures. To address this, we introduce standard-content characters as structural priors and explicitly model handwritten style cues as high-level residuals between handwritten references and their standard-content counterparts. This formulation suppresses shared content structures while emphasizing writer-specific deviations, such as stroke deformation, slant, curvature, spacing, and ligature patterns. Inspired by recent style-discriminative learning studies (Pan et al., 2026; Kotani et al., 2020), we further apply a feature-level writer classification loss to enhance the identity discriminability of the learned style representation.

Specifically, we propose a style–content decoupling encoder E_{sty}, as illustrated in the upper-left part of fig. 4. For each handwritten style reference R_{sty}, its standard-content counterpart $R_{\text{sty}}^{\text{con}}$ is obtained by indexing our self-constructed standard font library. Then we feed them into a dual-stream ResNet block to obtain the hidden features h_{sty} and $h_{\text{sty}}^{\text{con}}$. We then compute their feature-level residual as a style cue and use it to modulate both streams:

$$h_{\text{diff}} = \left|h_{\text{sty}} - h_{\text{sty}}^{\text{con}}\right|, \quad \tilde{h}_{\text{sty}} = \text{Conv}_1\left(h_{\text{diff}}\right) \odot h_{\text{sty}}, \quad \tilde{h}_{\text{sty}}^{\text{con}} = \text{Conv}_2\left(h_{\text{diff}}\right) \odot h_{\text{sty}}^{\text{con}}. \tag{1}$$

Here, Conv_1 and Conv_2 denote convolutional modulation layers, and $\odot$ denotes element-wise multiplication. This residual decoupling process is stacked 4 times to progressively refine the two streams, the final style feature of the k-th reference character is derived from the residual between the refined streams, i.e., $f_{\text{sty}}^{k} = \text{ResBlock}(|\tilde{h}_{\text{sty}} - \tilde{h}_{\text{sty}}^{\text{con}}|)$. The style features from all K reference characters are then concatenated along the temporal dimension to form the overall writer-level style representation f_{sty}.

Style–Content-Aware SOS Generator. Since online handwriting is generated auto-regressively, the initial decoding state directly affects the subsequent trajectory evolution. A fixed SOS token is independent of the target content and reference style, and thus provides limited guidance for generation. To obtain a more informative initialization, we design an SOS generator that performs interactive cross-attention between the

style feature f_{sty} and the content feature f_{con}. The generated style–content-aware SOS serves as a global control signal for the decoder, stabilizing the trajectory generation process.

Style–Content Conditional Decoder. The decoder takes the writer-style feature f_{sty} and content feature f_{con} as conditions, and generates online handwriting trajectories in an auto-regressive manner. Built upon causal self-attention, it first models temporal dependencies among previously generated points, and then incorporates writer style and target content through two independent cross-attention layers over f_{sty} and f_{con}, respectively. Before decoding, the style–content-aware SOS token is used to initialize the generation process, providing an adaptive starting state conditioned on both writer style and target content. This design enables each decoding step to be jointly guided by the historical trajectory, signer-specific writing patterns, and target character structure. At the output layer, a Gaussian Mixture Model (GMM) head predicts the distribution of two-dimensional coordinates, while a pen-state (PS) head estimates discrete pen states, enabling continuous trajectory generation with valid stroke transitions.

Training Objective. Similar to prior works (Dai et al., 2023; Tang & Lian, 2021), the reconstruction objective consists of a coordinate loss $\mathcal{L}_{\text{xy}}$ and a pen-state loss $\mathcal{L}_{\text{ps}}$. Given the ground-truth point $\mathbf{p}_t = (x_t, y_t)$, the coordinate loss is defined as the negative log-likelihood under the predicted GMM:

$$\mathcal{L}_{\text{xy}} = -\frac{1}{|\Omega|} \sum_{t \in \Omega} \log \left(\sum_{m=1}^{M} \pi_{m,t} \mathcal{N}(\mathbf{p}_t \mid \boldsymbol{\mu}_{m,t}, \boldsymbol{\Sigma}_{m,t}) \right), \tag{2}$$

where Ω denotes the set of valid non-padded trajectory steps, and M is the number of Gaussian mixture components. For the m-th component at step t, $\pi_{m,t}$ is the mixture weight, $\boldsymbol{\mu}_{m,t}$ is the two-dimensional mean, and $\boldsymbol{\Sigma}_{m,t}$ is the covariance matrix. The pen-state prediction is formulated as a binary classification problem, indicating whether the pen continues writing or terminates the current stroke.

Since the *Pen Down* state dominates online handwriting trajectories, we adopt focal loss (Lin et al., 2017) to alleviate the class-imbalance issue, as formulated in Eq. equation 3. Here L denotes the sequence length, and $y^{\text{pen}}_{t,k}$, $p^{\text{pen}}_{t,k}$ denote the ground-truth label and predicted probability of pen-state class k at time step t, respectively. Moreover, α_k is the class-balancing factor, and γ is the focusing parameter.

$$\mathcal{L}_{\text{ps}} = -\frac{1}{L} \sum_{t=1}^{L} \sum_{k=1}^{2} \alpha_k (1 - p^{\text{pen}}_{t,k})^{\gamma} y^{\text{pen}}_{t,k} \log p^{\text{pen}}_{t,k}, \tag{3}$$

Besides, we employ an inter-writer NCE loss on f_{sty} to encourage discriminative style representations :

$$\mathcal{L}_{\text{wri}} = -\frac{1}{\sum_{i,j} \mathbf{P}_{i,j}} \sum_{i,j} \mathbf{P}_{i,j} \log \frac{\exp(\text{sim}_{i,j})}{\exp(\text{sim}_{i,j}) + \sum_k (1 - \mathbf{P}_{i,k}) \exp(\text{sim}_{i,k})}, \tag{4}$$

where i, j, k index samples within a training batch, $\mathbf{P}_{i,j}$ is a boolean indicating whether sample i and j belong to the same writer, and $\text{sim}_{i,j}$ denotes the cosine similarity between the normalized style feature f_{sty} of sample i and j. The overall loss function is defined in eq. (5).

$$\mathcal{L} = \mathcal{L}_{\text{xy}} + 0.5 \cdot \mathcal{L}_{\text{ps}} + \mathcal{L}_{\text{wri}}. \tag{5}$$

3.3 Signature Synthesis for Verification

Signature Verification Task. Similar to the online handwritten character format defined in section 3.2, an online signature is represented as $\text{Sig} = \{p_1, p_2, \cdots, p_L\}$, where each point additionally includes pressure, i.e., $p_i = (x_i, y_i, pen_i, pre_i)$. In signature verification, samples are divided into **genuine signatures** $\mathcal{S}^{\text{real}}_{\text{g}}$, written by the target writer, and **forgery signatures** $\mathcal{S}^{\text{real}}_{\text{f}}$, written by others to imitate the target writer. A verifier is trained to distinguish genuine signatures from forgeries. However, in practical applications, forgery samples are often unavailable, and even genuine samples may be extremely limited. Therefore, we aim to build upon InkFormer as a signature synthesis engine to augment the training data for verification.

Signature Synthesis via InkFormer-Word. To enable InkFormer-Char to serve as an end-to-end data synthesis module, we fine-tune the InkFormer-Char using online word-level handwriting data consisting of 2–4

Chinese characters (as illustrated in the Stage II of fig. 2), resulting in **InkFormer-Word**. This adaptation equips the model with stable multi-character generation capability. Since each word-level reference contains richer style cues, we reduce the number of style references by half, from K to $K/2$.

During signature synthesis, real signature trajectories without pen pressure are provided as style references, together with the specified content prompts. InkFormer-Word then generates the corresponding online signature trajectories in a single forward pass. To reflect different practical conditions, flowing (Lai et al., 2022; Tolosana et al., 2021a), we simulate two representative real-world scenarios in verifier training, where both the available training data and the usage of synthesized signatures differ:

- **Forgery-free Verification.** In realistic signature verification, only several genuine samples $\mathcal{S}_{\mathrm{g}}^{\mathrm{real}}$ are usually available for each writer, while forgery signatures are absent. Since InkFormer-word disentangles content and style, we use the target writer's name as the content condition and a mixture of target-writer and other-writer signatures as style references to generate synthetic forged signatures $\mathcal{S}_{\mathrm{f}}^{\mathrm{syn}}$. This mixed-style conditioning simulates the forgery process, where an imitator attempts to reproduce the target writer's style while inevitably retaining part of their own writing style. These generated signatures serve as negative samples for verification training. Overall, after augmentation, each writer is associated with genuine signatures $\mathcal{S}_{\mathrm{g}}^{\mathrm{real}}$ and synthetic forged signatures $\mathcal{S}_{\mathrm{f}}^{\mathrm{syn}}$.
- **One-shot Genuine Verification.** In this extreme low-resource scenario, the genuine set of each writer contains only one real signature, denoted as $\mathcal{S}_{\mathrm{g:1}}^{\mathrm{real}}$. We synthesize two types of training samples. For genuine synthesis, the target writer's name and the only available genuine signature are used as the content condition and style reference, respectively, producing synthetic genuine signatures $\mathcal{S}_{\mathrm{g}}^{\mathrm{syn}}$ to augment positive samples. For forgery synthesis, the target writer's name is used as the content condition, while target-writer and other-writer signatures are jointly used as mixed style references, generating synthetic forged signatures $\mathcal{S}_{\mathrm{f}}^{\mathrm{syn}}$ to augment negative samples. Finally, the positive set is expanded to $\mathcal{S}_{\mathrm{g:1}}^{\mathrm{real}} \cup \mathcal{S}_{\mathrm{g}}^{\mathrm{syn}}$, while the negative set is constructed as $\mathcal{S}_{\mathrm{f}}^{\mathrm{syn}}$.

To reduce noise in the generated data and make the synthetic signatures suitable for verification training, we apply the following post-processing strategies: 1) **Quality filtering.** For each target signature, we generate multiple synthetic samples and retain only the top 20%. The selection is based on the DTW distance between each generated sequence and the corresponding real genuine signature, where samples with smaller DTW distances are considered higher-quality. 2) **Temporal resampling.** Since InkFormer is trained on trajectories preprocessed by the Ramer–Douglas–Peucker (RDP) algorithm (Douglas & Peucker, 1973), the synthesized signatures usually contain fewer points than real signatures. We therefore resample each synthesized trajectory to a length within 0.9–1.2 times that of the corresponding real signature. 3) **Pressure synthesis.** Since InkFormer predicts pen coordinates and pen states without pressure, we synthesize pressure values through DTW-based alignment. For each synthesized point s_{syn}^{i}, we find its aligned point set $\mathcal{M}(i)$ from a real genuine signature of the same writer and assign the average pressure: $p_{\mathrm{syn}}^{i} = \frac{1}{|\mathcal{M}(i)|} \sum_{s \in \mathcal{M}(i)} p(s)$.

4 Experiments

4.1 Dataset and Evaluation Metrics

Handwritten Character Dataset. To evaluate handwritten character generation, we use CASIA-OLHWDB 1.0–1.2 (Liu et al., 2011) for training and the ICDAR-2013 competition dataset (Yin et al., 2013) for testing. The training set contains 1,020 writers and approximately 6,700 Chinese character classes, with each writer providing about 3,755 samples on average. The test set includes 60 writers with character categories consistent with the training set. Following prior works, we simplify trajectory sequences using the Ramer–Douglas–Peucker (RDP) algorithm (Douglas & Peucker, 1973).

Handwritten Word Dataset. To evaluate the handwritten word generation capability, we construct a word-level dataset from CASIA-OLHWDB 2.0-2.2 (Liu et al., 2011), which comprises 1,019 writers and over 50,000 handwritten Chinese text-lines. We randomly segment the text lines into word samples of 2-4 characters, resulting in over 300,000 non-semantic handwritten words. We randomly select 919 writers for training and use the remaining 100 writers for testing.

Handwritten Signature Dataset. To evaluate the generated signatures for downstream signature verification, we use MSDS-ChS (Zhang et al., 2022), which contains Chinese signatures from 402 writers, with 20 genuine signatures and 20 skilled forgeries per writer. Following Zhang et al. (2022), the first 202 writers are used for training and the remaining 200 writers for testing.

Evaluation Metrics. We validate the proposed method from two aspects: its intrinsic generation capability and its effectiveness as a generative engine for improving downstream tasks. Specifically, we evaluate InkFormer from two complementary perspectives: generation quality and downstream verification utility.

For character- and word-level handwriting generation, we measure trajectory similarity using normalized DTW (nDTW)(Sakoe & Chiba, 1978; Chen et al., 2022). We also report Content Score (%) and Style Score (%)(Dai et al., 2023; Pan et al., 2026) to assess content fidelity and writer-style consistency, respectively. For signature generation, we evaluate the utility of synthesized signatures in downstream verification using DevInSight (Shi et al., 2025) as the backbone. Generated signatures are added to the training set under forgery-free and one-shot genuine settings, and the verifier is tested on real signatures. We report Equal Error Rate (EER) (Zhang et al., 2022) across skilled/random forgeries, global/local thresholds, and 4vs1/1vs1 template protocols. Each experiment is repeated three times, and averaged results are reported. More metric details are provided in section C.

Implementation Details. All decoder layers in InkFormer use an embedding dimension of 512, and the number of style references is set to $K = 15$ for character generation. InkFormer-Char is trained for 255,000 iterations with a batch size of 80, while InkFormer-Word is fine-tuned for 20,000 iterations using the same batch size. We adopt Adam with an initial learning rate of 1×10^{-4} and gradient clipping of 1.0. During fine-tuning, the learning rate is reduced to 5×10^{-5}. For signature verification, we follow the original training setting of DevInSight (Shi et al., 2025). For comparison methods, we preferentially use official pretrained weights when available; otherwise, we reproduce the results using the released code. All experiments are conducted on four NVIDIA RTX 2080 GPUs.

4.2 Main Results

4.2.1 Online Handwritten Character Generation

We compare InkFormer-Char with several representative online handwriting generation methods, including the auto-regressive approaches Deep Imitator (Zhao et al., 2020), Write Like You (Tang & Lian, 2021), and SDT (Dai et al., 2023), as well as the diffusion-based method OLHWG (Ren et al., 2025).

The qualitative results in fig. 5 demonstrate the effectiveness of our content and style modeling strategies. InkFormer-Char better preserves character structures while maintaining consistent writer-specific styles, producing more natural personalized handwriting patterns. This advantage is particularly evident in cases involving cursive connections and radical simplification, where existing methods often suffer from abnormal stroke transitions, inconsistent writing styles, or structural distortions. Additional visual comparisons are provided in fig. 10. Furthermore, as shown in table 1, InkFormer-Char achieves the best performance across all evaluation metrics. It reduces nDTW from 1.286 to 1.171 compared with SDT, indicating more accurate trajectory-level generation. For content fidelity, InkFormer-Char obtains the highest Content Score of 98.20%, outperforming SDT by 0.61% and OLHWG by 1.71%. This improvement benefits from the proposed modality-consistent content guidance, where online standard characters are used as content references, enabling the model to better preserve character structures while avoiding the alignment cost caused by cross-modality inputs. More importantly, InkFormer-Char achieves a Style Score of 83.78%, yielding a relative improvement of 28.66% over SDT and 37.90% over OLHWG. This significant gain demonstrates the effectiveness of our character-independent residual style modeling, which explicitly extracts writer-level style cues rather than relying only on loss-based style supervision.

4.2.2 Online Handwritten Word Generation

Since there are no directly comparable word-level online handwriting generation baselines, we construct several representative comparisons. First, we fine-tune SDT (Dai et al., 2023), the second-best character-level baseline, on the same word-level training data as InkFormer-Word, denoted as SDT-Word. Second, we

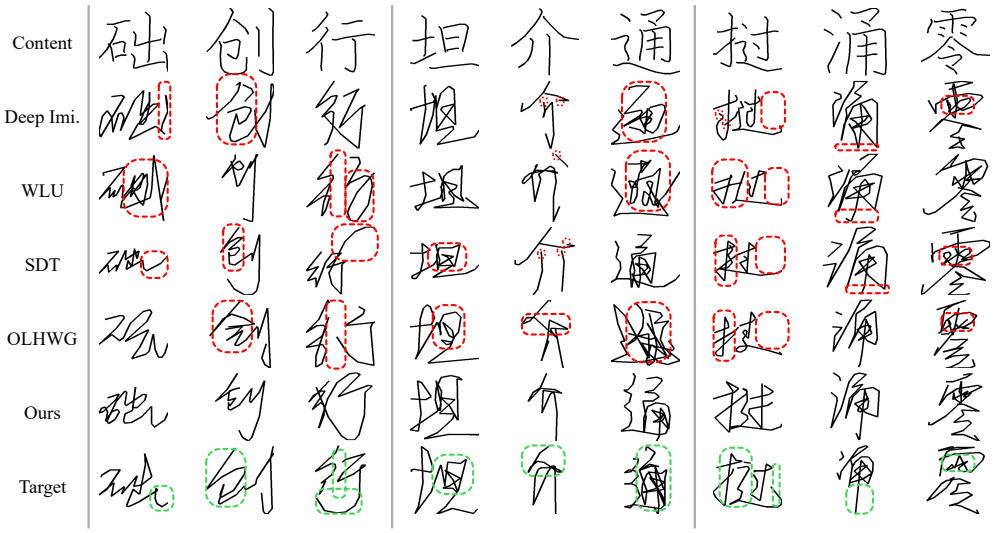


Figure 5: Qualitative comparison between InkFormer-Char and existing methods on **online handwriting character generation**. Red dashed boxes highlight failures in style imitation or structural preservation, while green dashed boxes indicate the target handwriting. Best viewed in color.

Table 1: Quantitatively comparing InkFormer-Char with existing methods on **online handwriting character generation**. The unit for nDTW metric is 0.001. The best results are marked in **bold**.

Method	Venue	nDTW↓	Content Score↑	Style Score↑
Deep Imitator (Zhao et al., 2020)	PR'20	1.559	92.13	46.04
Write Like You (Tang & Lian, 2021)	CGF'21	1.419	94.38	58.01
SDT (Dai et al., 2023)	CVPR'23	1.286	97.59	65.12
OLHWG (Ren et al., 2025)	ICLR'25	1.343	96.49	60.76
InkFormer-Char	Ours	**1.171**	**98.20**	**83.78**

include DiffInk (Pan et al., 2026), a diffusion-based line-level handwriting generation model. Third, following the two-stage character-layout generation paradigm (Ren et al., 2025), we combine character-level generators with an independent layout generator, resulting in SDT+Layout and InkFormer-Char+Layout.

The qualitative comparisons in fig. 6 show that InkFormer-Word better preserves writer-specific stroke styles while maintaining accurate character structures. Compared with character-wise generation or line-level baselines, InkFormer-Word produces smoother inter-character transitions and more coherent word-level layouts, especially around character junctions where stroke continuity and spacing are critical. These results demonstrate that end-to-end word-level modeling helps capture both intra-character structures and inter-character dependencies, leading to more realistic handwritten words. Additional visual comparisons are provided in fig. 11. As shown in table 2, InkFormer-Word achieves the best performance across all evaluation metrics, with an nDTW of 1.098, AR of 98.92%, CR of 98.98%, and Style Score of 93.29%. Compared with the two-stage Char+Layout version, InkFormer-Word further reduces nDTW from 1.257 to 1.098 and improves Style Score from 57.33% to 93.29%, demonstrating the advantage of end-to-end word-level modeling in capturing inter-character dependencies and maintaining consistent writing dynamics. Moreover, InkFormer consistently outperforms SDT under both Char+Layout and Word settings, confirming the effectiveness of the proposed content guidance and residual style modeling strategies. Compared with DiffInk, InkFormer-Word achieves much higher AR/CR and Style Score, indicating stronger structural accuracy and writer-style preservation for word-level online handwriting generation.

4.2.3 Online Handwritten Signature Synthesis

We compare InkFormer-Word with several signature synthesis baselines, including the re-parameterization-based Sigma-LogNormal method (Lai et al., 2022), DeepWriterSyn (Tolosana et al., 2021a), and the gen-

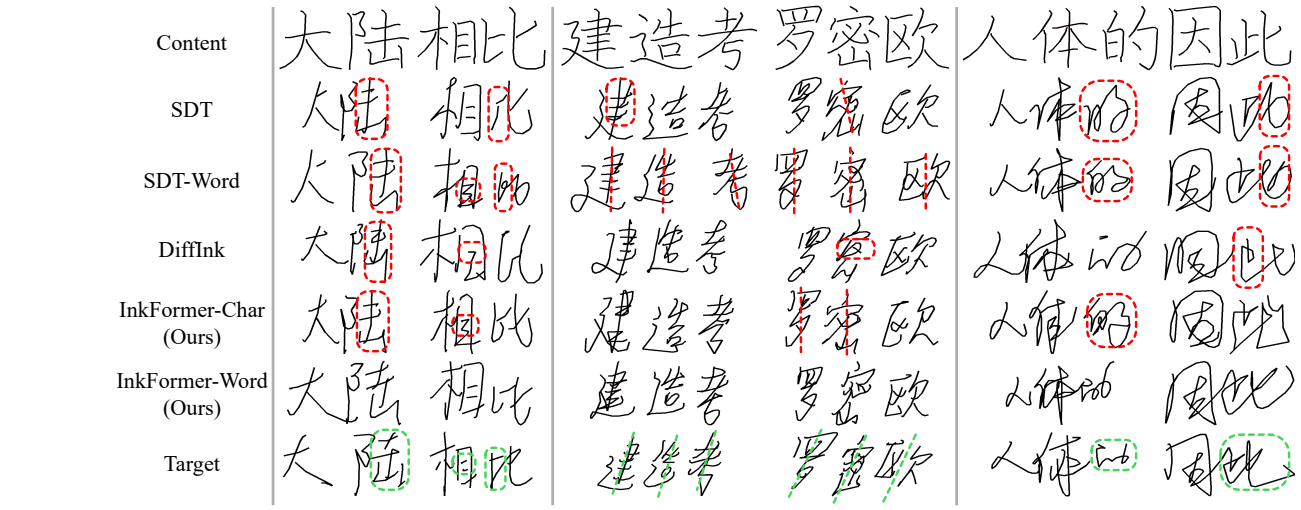


Figure 6: Qualitative comparison between InkFormer-Word and existing methods on **online handwriting word generation**. Red dashed boxes highlight failures in style imitation or structural preservation, while green dashed boxes indicate the target handwriting. Best viewed in color.

Table 2: Quantitatively comparing InkFormer-Word with existing methods on **online handwriting word generation**. The Content Score metric is changed to AR and CR (Pan et al., 2026). The unit for nDTW metric is 0.001. The best results are marked in **bold**.

Method	Venue	Granularity	nDTW↓	AR ↑	CR ↑	Style Score↑
SDT (Dai et al., 2023)	CVPR'23	Char+Layout	1.507	92.72	92.80	25.04
SDT-Word (Dai et al., 2023)	CVPR'23	Word	1.235	96.38	96.90	36.77
DiffInk (Pan et al., 2026)	ICLR'26	Line	1.359	92.54	94.49	80.75
InkFormer-Char	Ours	Char+Layout	1.257	98.20	98.27	57.33
InkFormer-Word	Ours	Word	**1.098**	**98.92**	**98.98**	**93.29**

erative baseline SDT-Word (Dai et al., 2023). We also include a no-synthesis baseline, where the verifier is trained only with random forgeries and no synthetic data. For synthesis-based methods, the generated signatures are used to augment verifier training under two practical scenarios. In the forgery-free setting, we synthesize forgery samples with a 1:1 ratio to the real genuine set. In the one-shot genuine setting, we first synthesize genuine samples to expand the original genuine set, and then synthesize forgery samples with a 1:1 ratio to the augmented genuine set. For each case, we initially generate five times the target number of samples and apply the filtering strategy section B to retain the final training samples. The results are reported in table 3, and visual examples are shown in fig. 7.

As illustrated in fig. 7, our approach explicitly models the imitation process rather than relying on fixed transformation patterns, resulting in more diverse and non-template-like samples. Additional visual results are provided in figs. 12 and 13. As shown in table 3, both re-parameterization-based baselines perform worse than no-synthesis baseline under the forgery-free training scenario, suggesting that when sufficient genuine samples are available, existing signature synthesis approaches may introduce considerable biased variation patterns that impair the generalization ability of the verifier. In contrast, under the more data-scarce one-shot genuine scenario, these baselines outperform no-synthesis baseline, indicating that synthetic samples can provide complementary benefits in extremely low-resource conditions. A similar trend is also observed in SDT-Word, a generative baseline, suggesting that insufficient style controllability leads to synthetic signatures with limited variation patterns. Compared to these methods, our approach consistently achieves superior performance in both training scenarios, outperforming both the no-synthesis baseline and existing signature synthesis methods, thereby demonstrating the stability and effectiveness of the proposed generation framework in realistic low-resource settings.

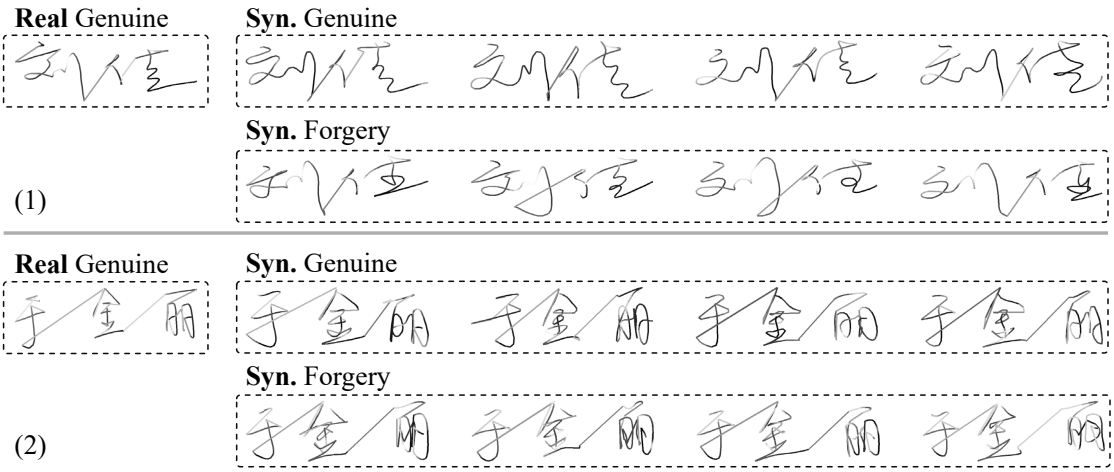


Figure 7: Visualization of real signatures and their corresponding synthetic counterparts generated by the proposed method. Both genuine and forgery samples are shown. Pen pressure is presented by color intensity, where darker color indicates higher pressure.

Table 3: Impact of **synthetic signatures** on downstream verification under forgery-free and one-shot genuine scenarios. We report EER (%) with global/local thresholds for skilled forgery and random forgery under 4 vs 1 and 1 vs 1 template settings. Only cross-session results are included.

Scenario	Method	Skilled Forgery ↓		Random Forgery ↓	
		4 vs 1	1 vs 1	4 vs 1	1 vs 1
Forgery-free	w/o synthesis	9.65/5.63	13.40/7.05	0.37/0.06	**0.38**/0.08
	Sigma-LogNormal (Lai et al., 2022)	10.79/7.27	15.22/8.63	0.50/**0.03**	0.41/**0.07**
	DeepWriterSyn (Tolosana et al., 2021a)	9.94/6.72	14.53/7.64	**0.34**/0.14	0.39/0.15
	SDT-Word (Dai et al., 2023)	10.73/7.36	14.88/8.42	0.59/0.12	0.47/0.15
	Ours	**8.99/5.57**	**12.76/6.71**	0.49/0.06	0.60/0.10
One-shot genuine	w/o synthesis	14.78/10.01	17.76/11.81	1.19/0.29	1.08/0.30
	Sigma-LogNormal	11.59/7.80	16.93/10.46	1.93/0.62	2.90/1.18
	DeepWriterSyn	11.94/7.96	17.32/11.25	1.31/0.47	2.75/1.52
	SDT-Word	12.77/8.71	17.78/10.20	0.66/0.23	0.69/0.31
	Ours	**10.80/7.40**	**14.85/8.69**	**0.59/0.20**	**0.49/0.20**

4.3 Ablation Study

Ablation on InkFormer-Char. As shown in table 4, each component contributes to the final performance of InkFormer-Char. Online content modeling improves structural accuracy by providing trajectory-level content guidance in the same modality as the generation target, while the writer-consistency constraint $\mathcal{L}_{\text{wri}}$ regularizes the style space by encouraging samples from the same writer to have closer representations. The most significant gain comes from the style–content decoupled encoder E_{sty}, suggesting that content-style entanglement is a major bottleneck in reference-based handwriting generation. By modeling the residual difference between handwritten references and standard-content characters, E_{sty} suppresses shared character structures and emphasizes writer-specific variations, leading to more stable and transferable style representations. The SOS generator further improves auto-regressive decoding by providing a style–content-aware initialization.

Signature Post-processing Ablation. As shown in table 5, we further analyze the impact of post-processing pipelines on downstream signature verification. Since InkFormer mainly generates trajectory coordinates and pen states, while real signature data usually contain higher temporal resolution and pressure information, directly using raw generated samples may introduce distribution gaps. Compared with the baseline that only includes length upsampling and naive pressure synthesis, introducing Quality Ranking consistently reduces the EER under skilled forgery settings, indicating that DTW-based quality filtering can

Table 4: Ablation Study on InkFormer-Char. In addition to quantitative results, several qualitative examples are presented. The unit of nDTW metric is 0.001.

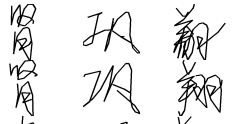
Method	Example	nDTW↓	Content Score↑	Style Score↑
Baseline		1.491	92.62	26.08
+ Online Cont.		1.308	97.21	38.73
+ $\mathcal{L}_{wri}$		1.293	97.36	51.84
+ $E_{sty:con}$		1.176	98.10	77.74
+ SOS Gen.		1.171	98.20	83.78
Ground Truth				

Table 5: Ablation study on post-processing pipelines in forgery-free training scenario. Baseline only contains length upsampling and naive pressure synthesis. Only cross-session results are shown.

Method	Skilled Forgery ↓		Random Forgery ↓	
	4 vs 1	1 vs 1	4 vs 1	1 vs 1
Baseline	9.39/5.83	13.01/6.93	0.54/0.08	0.61/0.10
+ Quality Ranking	9.10/5.64	12.97/6.83	0.54/0.06	0.62/0.11
+ DTW-based Pres. Syn.	8.99/5.57	12.76/6.71	0.49/0.06	0.60/0.10

remove low-quality samples with large trajectory deviations and retain synthetic signatures closer to the real data distribution. Based on this, DTW-based pressure synthesis further improves performance, suggesting that pressure is an important dynamic cue for signature verification and can be effectively transferred from real signatures to synthetic samples through DTW alignment. Overall, quality filtering and pressure synthesis improve both the trajectory quality and dynamic characteristics of synthetic signatures, making them more effective as training data for verification.

5 Conclusion

In this paper, we presented **InkFormer**, a controllable online handwriting generation framework for character-, word-, and signature-level synthesis. By introducing residual style–content decoupling, modality-consistent content conditioning, and progressive transfer from character-level handwriting to signature generation, InkFormer effectively captures writer-specific style while maintaining faithful content structures. Extensive experiments demonstrate that InkFormer achieves superior generation quality across different handwriting granularities. More importantly, when used as a generative engine for data augmentation, the synthesized signatures consistently improve downstream verification performance under both forgery-free and one-shot genuine training scenarios. These results suggest that controllable online handwriting generation can go beyond visual synthesis and serve as a practical data-centric solution for low-resource signature verification, offering a promising direction for generative modeling in biometric authentication.

Ethical Considerations. Although our signature synthesis method can significantly improve verification performance in low-data scenarios, it may also raise ethical risks, such as potential misuse for fraud. To mitigate these risks, all experiments are conducted on public or authorized datasets, and the generated signatures are used only for research purposes. We believe that any deployment of signature generation technologies should involve strict access control, monitoring mechanisms, and legal safeguards to prevent misuse. This work focuses on positive applications in authentication robustness and data augmentation, rather than signature forgery.

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.

S. D. Bhavani and R. K. Bharathi. A multi-dimensional review on handwritten signature verification: strengths and gaps. *Multimedia Tools and Applications*, 83(1):2853–2894, 2024. ISSN 1573-7721.

Weiran Chen, Guiqian Zhu, Ying Li, Yi Ji, and Chunping Liu. Advancements in chinese font generation since deep learning era: A survey. *arXiv preprint arXiv:2508.06900*, 2025.

Zhounan Chen, Daihui Yang, Jinglin Liang, Xinwu Liu, Yuyi Wang, Zhenghua Peng, and Shuangping Huang. Complex handwriting trajectory recovery: Evaluation metrics and algorithm. In *Proceedings of the asian conference on computer vision*, pp. 1060–1076, 2022.

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*, pp. 19054–19064, 2025.

Ayan Das, Yongxin Yang, Timothy Hospedales, Tao Xiang, and Yizhe Song. BézierSketch: A generative model for scalable vector sketches. In *European conference on computer vision*, pp. 632–647. Springer, 2020.

Ayan Das, Yongxin Yang, Timothy Hospedales, Tao Xiang, and Yizhe Song. ChiroDiff: Modelling chirographic data with diffusion models. *arXiv preprint arXiv:2304.03785*, 2023.

Moises Diaz, Andreas Fischer, Miguel A Ferrer, and Réjean Plamondon. Dynamic signature verification system based on one real signature. *IEEE transactions on cybernetics*, 48(1):228–239, 2016.

Moises Diaz, Andrea Mendoza-García, Miguel A Ferrer, and Robert Sabourin. A survey of handwriting synthesis from 2019 to 2024: A comprehensive review. *Pattern Recognition*, 162:111357, 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.

Miguel A Ferrer, Moises Diaz, Cristina Carmona-Duarte, and Réjean Plamondon. Generating off-line and on-line forgeries from on-line genuine signatures. In *2019 international Carnahan conference on security technology (ICCST)*, pp. 1–6. IEEE, 2019.

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*, pp. 16015–16025, 2025.

Alex Graves. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013.

David Ha and Douglas Eck. A neural representation of sketch drawings. *arXiv preprint arXiv:1704.03477*, 2017.

Yuan Hai, Zhang Heng, and Yin Fei. Online handwritten signature verification based on temporal-spatial graph attention transformer. *ArXiv*, abs/2510.19321, 2025. URL `https://api.semanticscholar.org/CorpusID:282272835`.

Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for MobileNetV3. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1314–1324, 2019.

Jiajia Jiang, Songxuan Lai, Lianwen Jin, and Yecheng Zhu. DsDTW: Local representation learning with deep soft-DTW for dynamic signature verification. *IEEE Transactions on Information Forensics and Security*, 17:2198–2212, 2022.

Harmandeep Kaur and Munish Kumar. Signature identification and verification techniques: state-of-the-art work. *Journal of Ambient Intelligence and Humanized Computing*, 14(2):1027–1045, 2023. ISSN 1868-5145.

Atsunobu Kotani, Stefanie Tellex, and James Tompkin. Generating handwriting via decoupled style descriptors. In *European Conference on Computer Vision*, pp. 764–780. Springer, 2020.

Songxuan Lai, Lianwen Jin, Yecheng Zhu, Zhe Li, and Luojun Lin. SynSig2Vec: Forgery-free learning of dynamic signature representations by sigma lognormal-based synthesis and 1D CNN. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6472–6485, 2022.

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.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017.

Chenglin Liu, Fei Yin, Dahan Wang, and Qiufeng Wang. CASIA online and offline chinese handwriting databases. In *2011 international conference on document analysis and recognition*, pp. 37–41. IEEE, 2011.

Chenglin Liu, Lianwen Jin, Xiang Bai, Xiaohui Li, and Fei Yin. Frontiers of intelligent document analysis and recognition: Review and prospects. *Journal of Image and Graphics*, 28(08):2223–2252, 2023.

Yitian Liu and Zhouhui Lian. DeepCalliFont: Few-shot chinese calligraphy font synthesis by integrating dual-modality generative models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 3774–3782, 2024.

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.

Konstantina Nikolaidou, George Retsinas, Giorgos Sfikas, and Marcus Liwicki. DiffusionPen: Towards controlling the style of handwritten text generation. In *European Conference on Computer Vision*, pp. 417–434. Springer, 2024.

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*, 2026. URL `https://openreview.net/forum?id=XKOEQFKFdL`.

Anurag Pandey, Pushap Deep Singh, Arnav Bhavsar, Aditya Nigam, Divya Acharya, and Basu Verma. Sign-Diffusion: Generating user specific online signature for digital verification. In *International Conference on Pattern Recognition*, pp. 31–45. Springer, 2024.

Anurag Pandey, Arnav Bhavsar Vinayak, Aditya Nigam, Divya Acharya, Basu Verma, et al. SIGN-GAIL: Rewarding online signature generation for digital imitation. In *Proceedings of the Winter Conference on Applications of Computer Vision*, pp. 1382–1391, 2025.

Vittorio Pippi, Fabio Quattrini, Silvia Cascianelli, Alessio Tonioni, and Rita Cucchiara. Zero-shot styled text image generation, but make it autoregressive. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 7910–7919, 2025.

Réjean Plamondon. A kinematic theory of rapid human movements: Part i. movement representation and generation. *Biological cybernetics*, 72(4):295–307, 1995.

Minsi Ren, Yanming Zhang, Qiufeng Wang, fei Yin, and Chenglin 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, Yanming 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`.

Hiroaki Sakoe and Seibi Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(1):43–49, 1978.

Yilin Shi, Lei Jiang, Hao Ni, and Lianwen Jin. DevInSight: Weaving path development into online signature verification. In *2025 International Conference on Document Analysis and Recognition (ICDAR)*, pp. 398–415. IEEE, 2025.

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, pp. 141–151. Wiley Online Library, 2021.

Ruben Tolosana, Paula Delgado-Santos, Andres Perez-Uribe, Ruben Vera-Rodriguez, Fierrez Julian, and Morales Aythami. DeepWriteSYN: On-line handwriting synthesis via deep short-term representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 600–608, 2021a.

Ruben Tolosana, Ruben Vera-Rodriguez, Julian Fierrez, and Javier Ortega-Garcia. DeepSign: Deep on-line signature verification. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 3(2):229–239, 2021b.

Honglie Wang, Minsi Ren, Yan-Ming Zhang, Fei Yin, and Chenglin Liu. Template-guided cascaded diffusion for stylized handwritten chinese text-line generation. In *International Conference on Document Analysis and Recognition*, pp. 149–166. Springer, 2025a.

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

Lei Wang, Yu Liu, Mohd Yunus Sharum, Razali Yaakob, Khairul Azhar Kasmiran, and Cunrui Wang. Deep learning for chinese font generation: A survey. *Expert Systems with Applications*, 276:127105, 2025b.

Lei Wang, Cunrui Wang, and Yu Liu. EHW-Font: A handwriting enhancement approach mimicking human writing processes. *Expert Systems with Applications*, 278:127278, 2025c.

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.

Hewei Yu and Pengfei Shi. A novel deep ensemble framework for online signature verification using temporal and spatial representation. In *International Conference on Information and Communications Security*, pp. 534–549. Springer, 2023.

Peirong Zhang, Jiajia Jiang, Yuliang Liu, and Lianwen Jin. MSDS: A large-scale chinese signature and token digit string dataset for handwriting verification. In *Advances in Neural Information Processing Systems*, volume 35, pp. 36507–36519, 2022.

Xuyao Zhang, Fei Yin, Yanming Zhang, Chenglin 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.

Bocheng Zhao, Jianhua Tao, Minghao Yang, Zhengkun Tian, Cunhang Fan, and Ye Bai. Deep imitator: Handwriting calligraphy imitation via deep attention networks. *Pattern Recognition*, 104:107080, 2020.

Appendix

The Appendix is structured as follows.

- In section A, we describe the construction of the **online content dataset** used for training.
- In section B, we introduce the **pre-processing pipeline** applied to the raw handwriting trajectories.
- In section C, we provide detailed definitions and implementation details of the **evaluation metrics** used in our experiments.
- In section D, we report additional results on **computational efficiency and resource consumption**.
- In section E, we present additional **feature visualization results** using t-SNE to compare different generation methods.
- In section F, we provide **additional qualitative results** of generated handwriting samples.
- In section G, we report additional experimental results on **Chinese signature synthesis**.

A Online Content Dataset

To provide content prompts for the proposed generation framework, we construct an online content dataset based on the open-source *Make-Me-A-Hanzi* dataset[1]. This dataset contains 9,574 Chinese characters written in regular script (Kai), where each character is annotated with stroke order as well as keypoints such as stroke turning points and endpoints. Based on these annotations, we reconstruct the corresponding online handwriting trajectories by connecting the ordered keypoints according to the provided stroke sequences.

The resulting online character set covers all Chinese characters appearing in the CASIA dataset (Liu et al., 2011), but a few characters in the MSDS-ChS signature dataset (Zhang et al., 2022) are still missing. To address this, we construct the missing characters using a radical substitution strategy. Specifically, we select structurally similar characters from the standard set and replace the corresponding radicals to obtain the target characters, as illustrated in fig. 8. This is feasible because Chinese characters often share compositional structures and common radicals, allowing reasonable stroke structures to be preserved.

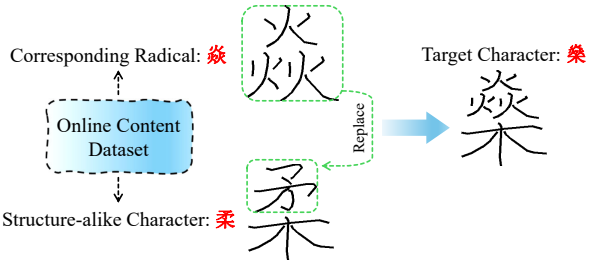


Figure 8: Example of constructing missing online standard characters by replacing radicals in structurally similar characters.

B Pre-processing Procedure

All generation tasks in this work share the same pre-processing pipeline. We first apply min-max normalization to the spatial coordinates. Due to the horizontally writing nature of handwriting, only the vertical range is normalized to 1.0, while the horizontal scale is adjusted according to the original aspect ratio. Following prior works, we then simplify handwriting trajectories using the Ramer-Douglas-Peucker (RDP) algorithm (Douglas & Peucker, 1973) with a tolerance of 0.2 to remove redundant points and reduce the length of trajectory samples. The online content trajectories are already compact and therefore bypass this step. Finally, absolute coordinates (x, y) are converted into relative velocities $(\Delta x, \Delta y)$.

[1] https://github.com/skishore/makemeahanzi

Table 6: Efficiency report of InkFormer on character, word and signature generation tasks.

Task	Avg Length	Speed
Character	33.90 pts	3.53 char/s
Word	104.48 pts	1.79 word/s
Signature	126.15 pts	0.48 sig/s

C Metric Details

Similar to prior works (Dai et al., 2023; Pan et al., 2026), we adopt three metrics: normalized Dynamic-Time-Warping (nDTW), style score, and content score.

nDTW. Dynamic Time Warping (DTW) (Sakoe & Chiba, 1978) is a widely used similarity measure for sequences with different lengths. Given a generated trajectory G and a target trajectory T, DTW computes the optimal alignment cost between the two sequences. To make the score comparable across samples, we normalize the DTW distance by the trajectory length and spatial scale:

$$\mathrm{nDTW}(G, T) = \frac{\mathrm{DTW}(G, T)}{L_T \cdot d_T},$$

where L_T denotes the length of the reference trajectory and d_T represents its spatial scale, defined as the diagonal length of the trajectory's bounding box.

Style Score. We measure the style consistency of generated samples using a writer identification model trained on the test set. The classifier adopts a ResNet18-1D backbone, whose classification head is adapted to the number of writers in the dataset and optimized using cross-entropy loss. The style score is defined as the average writer identification accuracy as section C, where a_i and $\hat{a}_i$ denote the ground-truth and predicted writer labels, respectively. $\mathrm{StyleScore} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}(\hat{a}_i = a_i)$.

To improve robustness in style evaluation, we concatenate 15 generated characters into a sequence and perform writer prediction jointly, while in word evaluation the number is reduced to 8. Different from prior works that rely on rendered offline images, we perform the evaluation directly in the online modality, which better preserves stylistic cues such as stroke order and writing dynamics. The trained classifier achieves accuracies of 99.97% and 99.99% on the character-level and word-level test sets, respectively.

Content Score. Content fidelity is measured using content recognizers. For character evaluation, we train a character classifier with a ResNet18-1D backbone and a classification head optimized using cross-entropy loss. The resulting classifier achieves an accuracy of 95.99% on the test set, and the average prediction accuracy is used as the content score.

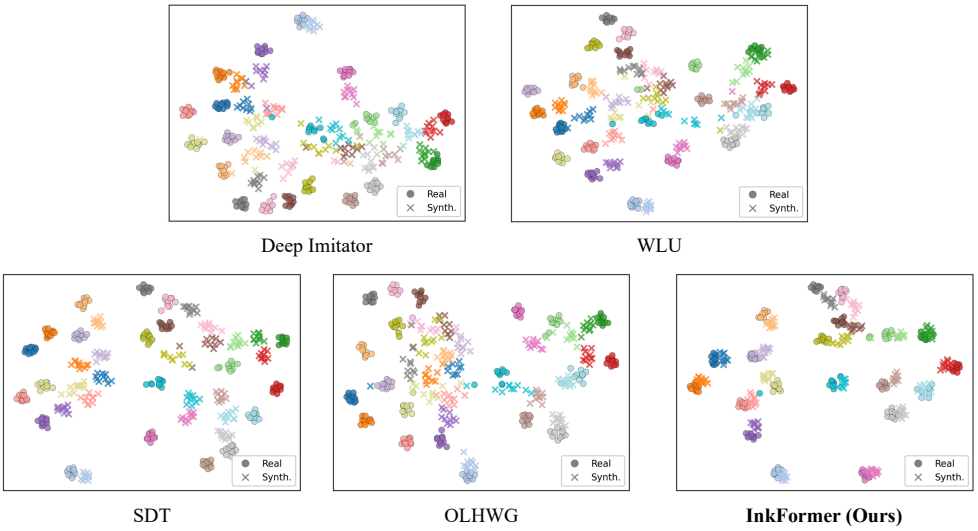
For word-level evaluation, we attach an LSTM decoder after the same backbone and train the model with Connectionist-Temporal-Classification (CTC) loss. The model achieves accuracy of 97.62%. Following standard handwriting recognition evaluation protocols (Liu et al., 2011), we report Accurate Rate (AR) and Correct Rate (CR) as content metrics:

$$AR = \frac{N_t - D_e - S_e - I_e}{N_t}, \qquad CR = \frac{N_t - D_e - S_e}{N_t}. \tag{6}$$

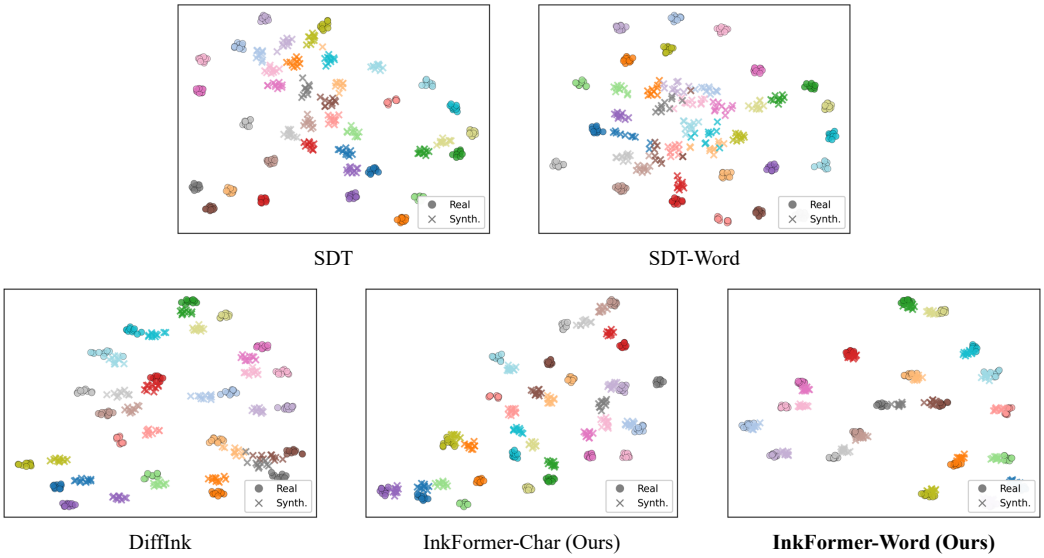
where N_t is the total number of ground-truth characters, and D_e, S_e, and I_e denote the numbers of deletion, substitution, and insertion errors, respectively.

D Efficiency Report

table 6 reports the generation efficiency of the proposed InkFormer for characters, words, and signatures. All results are measured on a single RTX 2080Ti GPU with a batch size of 1, without additional inference acceleration techniques(e.g. KV cache). The model contains approximately 36M parameters. These results demonstrate the feasibility of InkFormer for large-scale handwriting generation in practical scenarios.



(a) t-SNE visualization of generated online handwriting characters.



(b) t-SNE visualization of generated online handwriting words.

Figure 9: t-SNE visualizations of generated characters and words against real samples. Colors denote different writers, with circles for real samples and crosses for synthetic samples.

E t-SNE Comparison

We further present t-SNE visualizations of generated samples and real samples in the style feature spaces. Character-level and word-level results are show in fig. 9 (a) and (b) respectively. The style-level features are extracted from style score evaluating model. As shown in the figures, samples generated by InkFormer-Char and InkFormer-Word exhibit distributions that are more aligned with the real data, indicating better preservation of stylistic characteristics consistency.

F More Visualizations

fig. 10 and fig. 11 present additional qualitative comparisons between our method and the top-performing baseline SDT (Dai et al., 2023). The results show that our approach achieves better style fidelity and content accuracy for both character and word generation. We also provide additional visualizations of synthesized signatures produced by our framework in fig. 12 and fig. 13.

Figure 10: Additional visual comparisons of generated online handwritten **characters**.

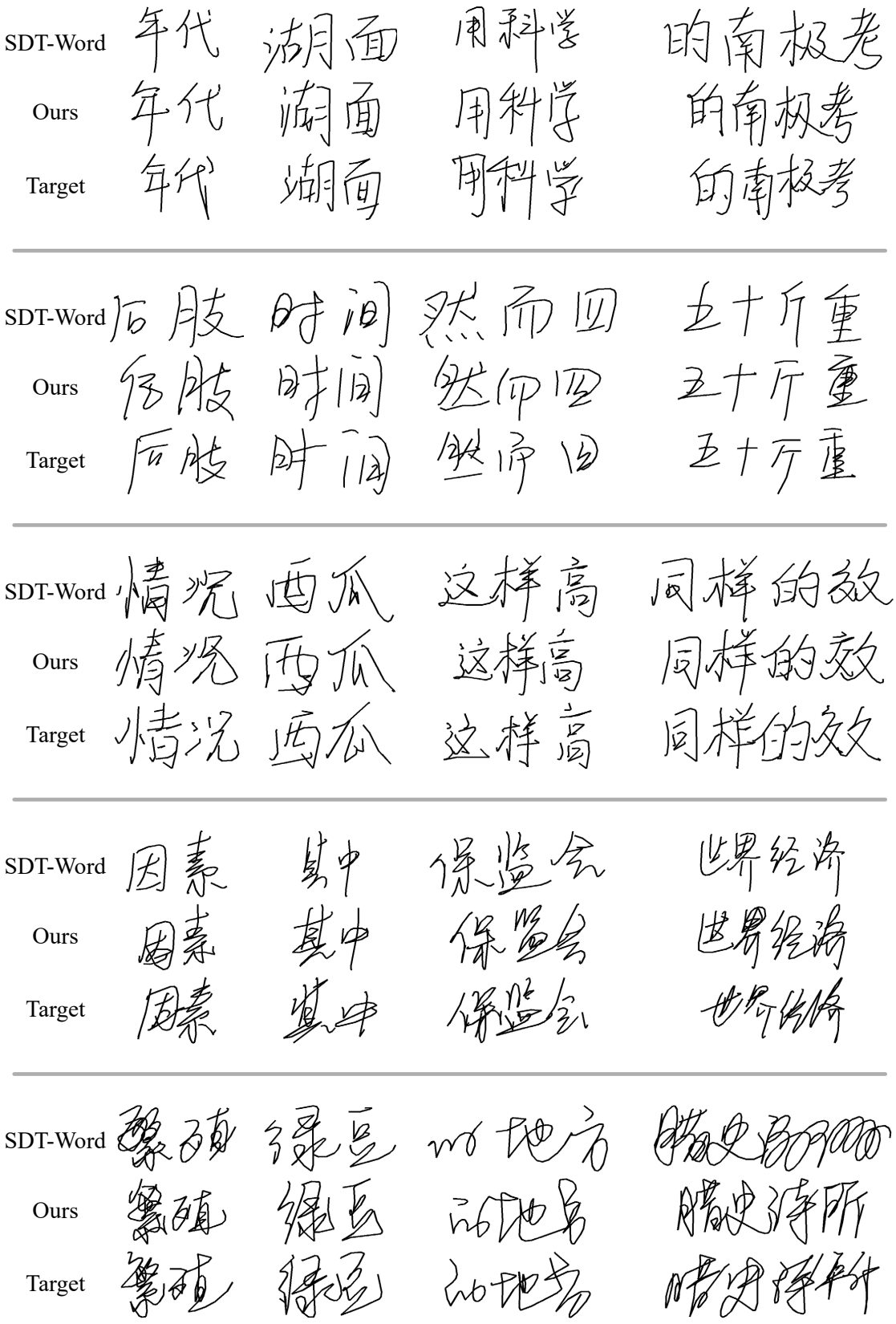


Figure 11: Additional visual comparisons of generated online handwritten **words**.

Figure 12: Additional visualization of the synthetic **signatures**.

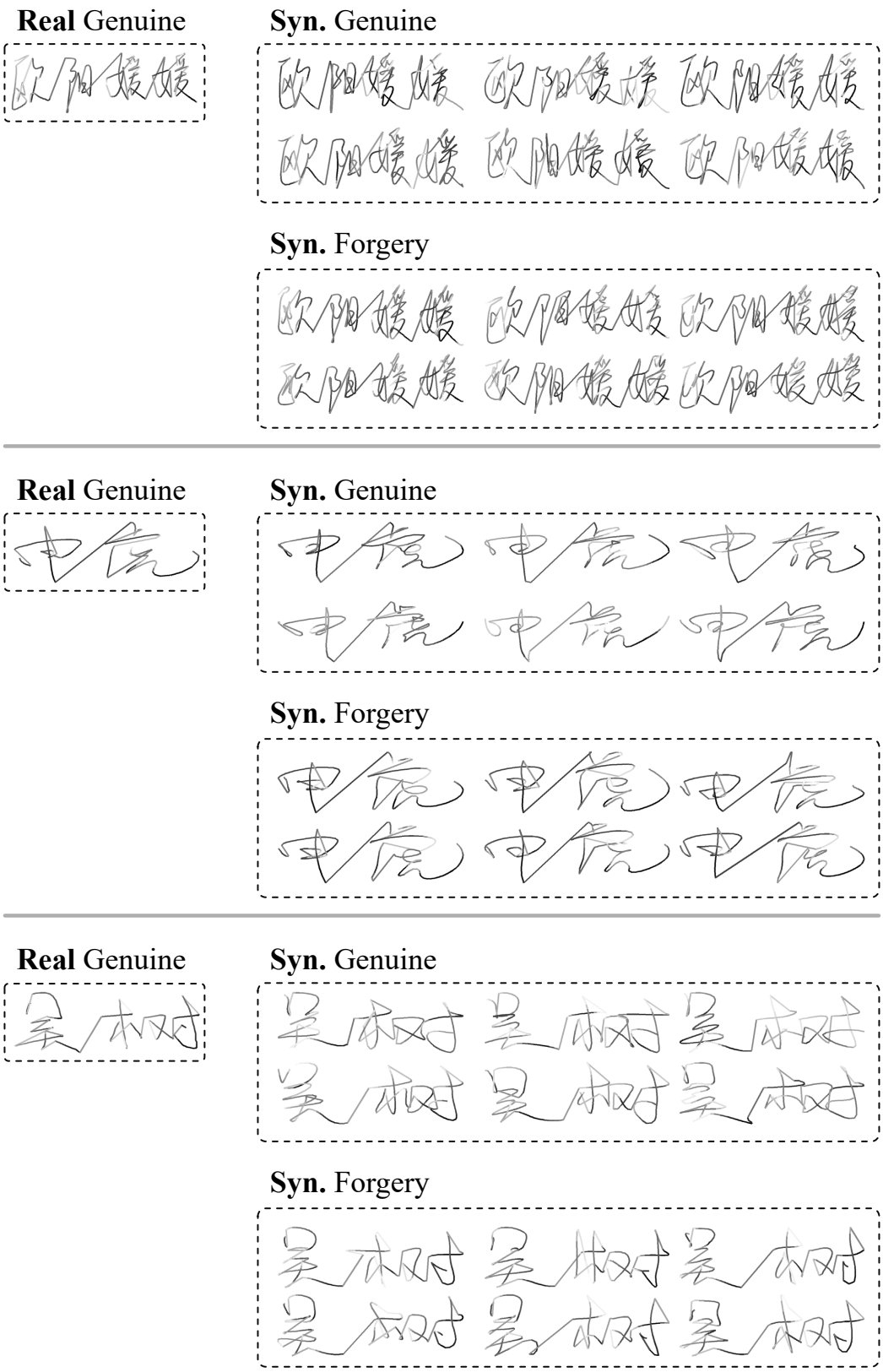


Figure 13: Additional visualization of the synthetic **signatures**.

Table 7: **Complete results** of online handwriting signature synthesis on enhancing **forgery-free training scenario**. EER(%) of global/local threshold, skilled and random forgery testing, different number of templates (4vs1 and 1vs1) are reported. The best results of each session are marked in **bold**.

Session	Method	Skilled Forgery ↓		Random Forgery ↓	
		4 vs 1	1 vs 1	4 vs 1	1 vs 1
1	w/o synthesis data	2.18/0.45	7.54/1.09	**0.00/0.00**	**0.01/0.00**
	Sigma-LogNormal (Lai et al., 2022)	3.33/0.68	8.98/1.41	**0.00/0.00**	0.02/**0.00**
	DeepWriterSyn (Tolosana et al., 2021a)	3.08/0.62	8.77/1.37	**0.00/0.00**	0.03/**0.00**
	SDT-Word (Dai et al., 2023)	3.45/0.92	10.07/2.15	0.08/**0.00**	0.10/**0.00**
	Ours	**1.73/0.46**	**6.52/0.74**	**0.00/0.00**	**0.01/0.00**
2	w/o synthesis data	2.96/0.29	8.82/1.27	**0.00/0.00**	**0.09/0.00**
	Sigma-LogNormal	3.53/0.53	11.07/1.50	**0.00/0.00**	0.12/**0.00**
	DeepWriterSyn	2.94/0.50	9.30/1.42	**0.00/0.00**	0.13/**0.00**
	SDT-Word	4.15/1.10	11.93/2.22	0.08/**0.00**	0.37/0.07
	Ours	**2.13/0.25**	**7.05/0.79**	**0.00/0.00**	0.16/**0.00**
1 & 2	w/o synthesis data	9.65/5.63	13.40/7.05	0.37/0.06	**0.38**/0.08
	Sigma-LogNormal	10.79/7.27	15.22/8.63	0.50/**0.03**	0.41/**0.07**
	DeepWriterSyn	9.94/6.72	14.53/7.64	**0.34**/0.14	0.39/0.15
	SDT-Word	10.73/7.36	14.88/8.42	0.59/0.12	0.47/0.15
	Ours	**8.99/5.57**	**12.76/6.71**	0.49/0.06	0.60/0.10

G More Experiments on Signature Synthesis

G.1 Complete Results Across Session Settings

We report the complete evaluation results for signature synthesis under both **single-session (1 or 2)** and **cross-session (1&2)** testing, using DevInSight (Shi et al., 2025) as the verifier backbone. Results for the forgery-free and one-shot genuine scenarios are shown in table 7 and table 8, respectively.

In the single-session setting, existing signature synthesis baselines fail to outperform the model trained without synthesized data, while their advantages become more evident in the more challenging cross-session, 1-shot genuine training setting. This phenomenon suggests that when intra-writer variation is relatively small, current synthesis methods may introduce template bias, which can misguide the training process. In contrast, under larger cross-session variations, the additional diversity provided by synthesized samples becomes more beneficial.

Overall, our approach consistently achieves superior performance in both training scenarios, outperforming both the no-synthesis baseline and existing signature synthesis methods, thereby demonstrating the stability and effectiveness of the proposed generation framework in realistic low-resource settings.

G.2 Alternative Verifier Evaluation

Besides using DevInSight (Shi et al., 2025) as the verifier backbone, we further evaluate our synthesized data with another strong verifier, DsDTW (Jiang et al., 2022). Results for the forgery-free and one-shot genuine scenarios are reported in table 9 and table 10, respectively. Under data-scarce scenarios, our method improves most evaluation metrics compared with training using only real data. These results highlight the potential of our synthesis framework as a general and effective data augmentation strategy for improving signature verification systems under data-scarce scenarios.

Table 8: **Complete results** of online handwriting signature synthesis on enhancing **1-shot genuine training scenario**. EER(%) of global/local threshold, skilled and random forgery testing, different number of templates (4vs1 and 1vs1) are reported. The best results of each session are marked in **bold**.

Session	Method	Skilled Forgery ↓		Random Forgery	
		4 vs 1	1 vs 1	4 vs 1	1 vs 1
1	w/o synthesis data	**1.97/0.24**	8.04/1.50	**0.00/0.00**	0.22/0.04
	Sigma-LogNormal (Lai et al., 2022)	3.60/0.85	10.29/3.16	0.05/**0.00**	0.76/0.14
	DeepWriterSyn (Tolosana et al., 2021a)	3.65/0.88	9.89/1.99	0.08/**0.00**	0.10/0.01
	SDT-Word (Dai et al., 2023)	3.63/1.05	10.55/2.02	0.08/**0.00**	0.13/0.03
	Ours	2.49/0.61	**7.97/1.44**	**0.00/0.00**	**0.01/0.00**
2	w/o synthesis data	2.91/0.85	8.95/2.37	0.05/0.01	0.27/0.02
	Sigma-LogNormal	3.95/1.09	11.43/3.27	0.14/0.03	1.00/0.17
	DeepWriterSyn	4.08/1.07	11.92/2.18	0.08/**0.00**	0.33/0.06
	SDT-Word	4.19/1.07	11.61/2.09	0.08/**0.00**	0.35/0.06
	Ours	**2.86/0.54**	**8.89/1.51**	**0.00/0.00**	**0.12/0.00**
1 & 2	w/o synthesis data	14.78/10.01	17.76/11.81	1.19/0.29	1.08/0.30
	Sigma-LogNormal	11.59/7.80	16.93/10.46	1.93/0.62	2.90/1.18
	DeepWriterSyn	11.94/7.96	17.32/11.25	1.31/0.47	2.75/1.52
	SDT-Word	12.77/8.71	17.78/10.20	0.66/0.23	0.69/0.31
	Ours	**10.80/7.40**	**14.85/8.69**	**0.59/0.20**	**0.49/0.20**

Table 9: Results of online handwriting signature synthesis on enhancing **DsDTW** in **forgery-free training scenario**. EER(%) of global/local threshold, skilled and random forgery testing, different number of templates (4vs1 and 1vs1) are reported. The best results of each session are marked in **bold**.

Session	Method	Skilled Forgery ↓		Random Forgery ↓	
		4 vs 1	1 vs 1	4 vs 1	1 vs 1
1	w/o synthesis data	3.00/**0.53**	7.66/1.78	**0.00/0.00**	0.10/0.00
	Ours	**2.83**/0.63	**7.42/1.63**	**0.00/0.00**	**0.06/0.00**
2	w/o synthesis data	2.35/0.35	7.30/1.54	**0.00/0.00**	**0.10/0.00**
	Ours	**2.23/0.27**	**7.28/1.48**	**0.00/0.00**	0.15/**0.00**
1 & 2	w/o synthesis data	11.46/8.12	14.67/9.59	0.83/0.03	0.77/0.13
	Ours	**10.81/7.53**	**14.57/9.32**	**0.74/0.02**	**0.70/0.10**

G.3 N-shot Genuine Training Scenario

We further assess our synthesized data in additional limited-data scenarios: 4-shot and 10-shot genuine training. Compared with the one-shot genuine scenario, these settings are less extreme and enable the verifier to better capture intra-writer variation during training. The results are shown in table 11 and table 12, respectively. In these two scenarios, the synthesized signatures generated by our method still improve the performance of the verification models, while the improvements are more pronounced in the single-session setting (session 1 or 2). This is mainly because the MSDS-ChS dataset (Zhang et al., 2022) contains signatures collected in two different periods, which introduces substantial intra-writer variations. In the current experimental setup, the verifier is trained only with data from session-1, and therefore cannot access the distribution of session-2 signatures or synthesize samples corresponding to that session. As a result, under the cross-session (1&2) testing protocol, where a larger session discrepancy exists between query and template signatures, the advantage brought by the synthesized signatures becomes less significant.

Table 10: Results of online handwriting signature synthesis on enhancing **DsDTW** in **one-shot genuine training scenario**. EER(%) of global/local threshold, skilled and random forgery testing, different number of templates (4vs1 and 1vs1) are reported. The best results of each session are marked in **bold**.

Session	Method	Skilled Forgery ↓		Random Forgery ↓	
		4 vs 1	1 vs 1	4 vs 1	1 vs 1
1	w/o synthesis data	2.85/**0.27**	8.92/2.30	0.17/0.00	0.50/0.12
	Ours	**2.58**/0.75	**8.22/1.88**	**0.00/0.00**	**0.02/0.00**
2	w/o synthesis data	3.00/0.55	9.35/1.87	0.08/0.08	0.59/0.13
	Ours	**2.42/0.38**	**8.66/1.76**	**0.00/0.00**	**0.14/0.00**
1 & 2	w/o synthesis data	13.23/9.51	17.08/11.11	2.78/0.72	2.95/1.29
	Ours	**12.89/9.14**	**16.66/10.97**	**0.90/0.10**	**0.77/0.20**

Table 11: Results of online handwriting signature synthesis on enhancing **4-shot genuine training scenario**. EER(%) of global/local threshold, skilled and random forgery testing, different number of templates (4vs1 and 1vs1) are reported. The best results of each session are marked in **bold**.

Session	Method	Skilled Forgery ↓		Random Forgery ↓	
		4 vs 1	1 vs 1	4 vs 1	1 vs 1
1	w/o synthesis data	2.03/0.48	7.18/**0.97**	**0.00/0.00**	**0.01/0.00**
	Ours	**1.82/0.35**	**6.84/0.97**	**0.00/0.00**	**0.01/0.00**
2	w/o synthesis data	2.67/0.33	8.56/1.18	**0.00/0.00**	**0.13/0.00**
	Ours	**1.93/0.28**	**7.51/0.86**	**0.00/0.00**	0.19/**0.00**
1 & 2	w/o synthesis data	10.41/6.54	13.66/7.85	**0.52**/0.14	**0.41**/0.15
	Ours	**10.20/6.47**	**13.59/7.65**	0.63/**0.08**	0.75/**0.11**

Table 12: Results of online handwriting signature synthesis on enhancing **10-shot genuine training scenario**. EER(%) of global/local threshold, skilled and random forgery testing, different number of templates (4vs1 and 1vs1) are reported. The best results of each session are marked in **bold**.

Session	Method	Skilled Forgery ↓		Random Forgery ↓	
		4 vs 1	1 vs 1	4 vs 1	1 vs 1
1	w/o synthesis data	2.18/0.53	7.59/1.15	**0.00/0.00**	0.01/**0.00**
	Ours	**1.81/0.32**	**6.66/0.96**	**0.00/0.00**	**0.00/0.00**
2	w/o synthesis data	2.72/0.36	9.06/1.39	**0.00/0.00**	**0.13/0.00**
	Ours	**2.06/0.23**	**7.36/0.89**	**0.00/0.00**	0.19/**0.00**
1 & 2	w/o synthesis data	10.49/6.59	13.91/7.81	**0.43**/0.11	**0.39**/0.13
	Ours	**9.88/6.29**	**13.78/7.72**	0.62/**0.09**	0.75/**0.11**