Domain-Agnostic Neural Oil Painting via Normalization Affine Test-Time Adaptation

Qichao Dong
University of Macau
Macau, China
mc35205@um.edu.mo

Lingyu Liu
Xi'an Jiaotong University
Xi'an, China
liulingyu@stu.xjtu.edu.cn

Yaxiong Wang
Hefei University of Technology
Hefei, China
wangyx@hfut.edu.cn

Jason J. R. Liu*
University of Macau
Macau, China
jasonliu@um.edu.mo

Zhedong Zheng*
University of Macau
Macau, China
zhedongzheng@um.edu.mo

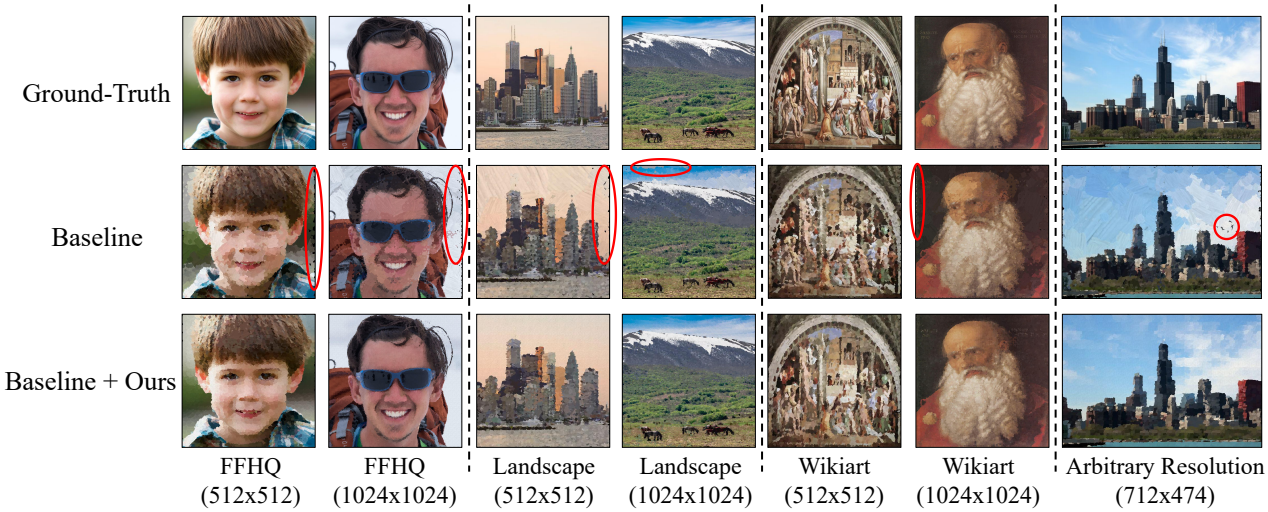

Figure 1: Painting comparison between the baseline method and ours on three real-world domains (*i.e.*, face, landmark and art photos). Due to the domain gap between the training set and real-world test images, existing methods usually suffer from over-smoothed textures and inconsistent brush granularity (highlighted in red circles). In contrast, the proposed test-time adaptation method efficiently adapts to the target scenario, further refining the visual quality across arbitrary input resolutions.

Abstract

Neural oil painting synthesis is to sequentially predict brushstroke color and position, forming an oil painting step by step, which could serve as a painting teacher for education and entertainment. Existing methods usually suffer from degraded generalization for real-world photo inputs due to the training-test distribution gap, often manifesting as stroke-induced artifacts (*e.g.*, over-smoothed textures or inconsistent granularity). In an attempt to mitigate this gap, we introduce a domain-agnostic neural painting (**DANP**) framework that aligns model to the test domain. In particular, we focus on updating affine parameters of normalization layers efficiently, while keeping other parameters frozen. To stabilize adaptation, our framework introduces: (1) Asymmetric Dual-Branch with mirror augmentation for robust feature alignment via geometric transformations, (2) Dual-Branch Interaction Loss combining intra-branch reconstruction and inter-branch consistency, and we also involve an empirical optimization strategy to mitigate gradient oscillations in practice. Experiments on real-world images from diverse domains (*e.g.*, faces, landscapes, and artworks) validate the effectiveness of DANP in resolution-invariant adaptation, decreasing ~11.3% reconstruction error at 512px and ~20.3% at 1024px compared to the baseline model. It is worthy noting that our method is compatible with existing methods, *e.g.*, Paint Transformer, and further improve the ~10.3% perceptual quality. Dataset and code will be publicly released at: https://domain-agnostic-neural-oil-painting.github.io/DANP.

Corresponding author.

MM '25, Dublin, Ireland.

ACM ISBN 979-8-4007-2035-2/2025/10
https://doi.org/10.1145/3746027.3758165

CCS Concepts

• **Applied computing** → **Media arts**; • **Computing methodologies** → **Image-based rendering**; **Unsupervised learning**.

Keywords

Test-Time Adaptation, Neural Painting, Unsupervised Domain Adaptation, Artistic Image Synthesis

ACM Reference Format:
Qichao Dong, Lingyu Liu, Yaxiong Wang, Jason J. R. Liu, and Zhedong Zheng. 2025. Domain-Agnostic Neural Oil Painting via Normalization Affine Test-Time Adaptation. In *Proceedings of the 33rd ACM International Conference on Multimedia (MM '25), October 27–31, 2025, Dublin, Ireland*. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3746027.3758165

1 Introduction

Neural painting techniques [12, 28, 28, 29, 35, 38], which enhance the artistic expressiveness of automated painting systems through parameterized brushstroke modeling and neural network optimization, have established themselves as pivotal tools for digital art synthesis [8, 22, 30, 32]. Capable of generating highly anthropomorphic artistic effects across portraiture, landscape, and abstract art domains, these methods significantly surpass conventional mechanical painting systems [11, 21, 33] in simulating authentic brushstroke patterns. Current research efforts predominantly concentrate on three methodological strands: reinforcement learning (RL)-based feedforward networks [13, 14, 30, 35], iterative brushstroke optimization [41], and Transformer-enabled parallel generation architectures [23].

However, two critical limitations persist across these approaches: **(1) Overreliance on training-test distribution consistency.** Liu *et al.* [23] proposed a self-training pipeline that formulates neural painting as a set prediction problem using a Transformer-based framework (Paint Transformer), enabling parallel stroke generation to significantly reduce inference time. While this approach enhances adaptability by eliminating the need for annotated datasets, it frequently succumbs to domain gaps when confronted with real-world images, resulting in brushstroke-content mismatches such as inconsistent texture granularity and boundary artifacts. **(2) Narrow focus on spatial stroke optimization.** Prevailing methods prioritize where to paint through stroke region optimization at the expense of cross-domain generalization. For instance, Hu *et al.* [13] trained an RL-based agent to dynamically determine painting regions. However, this strategy exhibits severely degraded stability when processing out-of-distribution images. Zou *et al.* [18, 41] developed an iterative brushstroke parameter search strategy to optimize stroke precision. Despite its improved accuracy, the method's excessive computational complexity renders it impractical for real-time deployment.

Traditional domain adaptation (DA) methods [7, 39], which necessitate simultaneous access to both source-domain (*e.g.*, synthetic canvases) and target-domain data alongside joint training via cross-domain loss functions [6, 27, 34, 37], prove inapplicable to practical artistic generation scenarios where source brushstroke data is unavailable post-deployment. While Test-Time Training (TTT) [31] supports single-domain adaptation, its reliance on coupled optimization of supervisory signals and self-supervised tasks risks [2, 25] compromising artistic style consistency—a critical requirement for preserving aesthetic integrity. In contrast, Test-Time Adaptation (TTA) [15, 19, 36, 40] operates as an unsupervised paradigm that dynamically adjusts model parameters using exclusively target-domain data during inference, eliminating dependencies on source data or labels. This capability positions TTA as a robust solution to heterogeneous data distribution challenges inherent in artistic synthesis, particularly in resolving discrepancies between real-world images and synthetic brushstroke data, such as (1) Illumination variances. Mismatches in lighting conditions between training simulations and real environments. (2) Material texture gaps. Divergences in surface reflectance properties of digitally simulated versus physical brushstrokes.

We present a domain-agnostic neural painting (**DANP**) framework, a novel framework that pioneers the integration of Test-Time Adaptation (TTA) into neural oil painting to resolve domain gaps between synthetic brushstroke distributions and real-world imagery. The framework's core innovation lies in its multi-stage adaptation architecture. First, input images undergo pyramidal hierarchical decomposition, where a multi-resolution pyramid structure enables progressive refinement—base layers prioritize boundary feature extraction to preserve fine-grained details, while subsequent layers iteratively enhance global stroke-texture coherence. Building upon this hierarchical representation, DANP implements parameter-efficient domain alignment by selectively updating the affine parameters of BatchNorm and LayerNorm layers, thereby maintaining the pretrained backbone's integrity while minimizing computational overhead. To stabilize adaptation under limited test data without any manual annotation, we introduce an asymmetric dual-branch architecture with geometry-aware mirror augmentation. This design enforces cross-view consistency through two complementary mechanisms. Complementing this approach, a progressive learning rate scheduler phases in normalization layer adjustments during initial adaptation steps, reducing statistical oscillation amplitudes and ensuring stable convergence. In summary, our contributions are threefold:

- **Domain-Agnostic Neural Painting (DANP) Framework:** We propose a new domain-agnostic framework for neural oil painting synthesis that mitigates the train-test distribution gap by efficiently adapting the model to the test domain. It is achieved through selective fine-tuning of the affine parameters in the normalization layers, while freezing the majority of parameters.
- **Selective Update of Normalization Affine Parameters via Asymmetric Dual-Branch Architecture & Dual-Branch Interaction Loss:** To ensure robust and stable adaptation to diverse real-world inputs, our framework introduces asymmetric mirror augmentation branches using geometric transformations to promote robust feature alignment, and a dual-branch interaction loss combining intra-branch reconstruction and inter-branch consistency constraints.
- **Significant Performance Gains & Compatibility:** Our method shows resolution-invariant adaptation, substantially reducing reconstruction error (11.3% at 512px, 20.3% at 1024px) and improving perceptual quality (10.3%) on diverse real-world images. Crucially, DANP is compatible with existing methods (*e.g.*, Paint Transformer and Comoisitional Neural Painter), further enhancing the visual quality.

2 Related Work

Neural Painting. Current stroke-based neural painting methods primarily revolve around deep learning frameworks, encompassing reinforcement learning, feedforward neural networks, and optimization techniques. Among these, the Compositional Neural Painter (based on reinforcement learning) [13] and the Paint

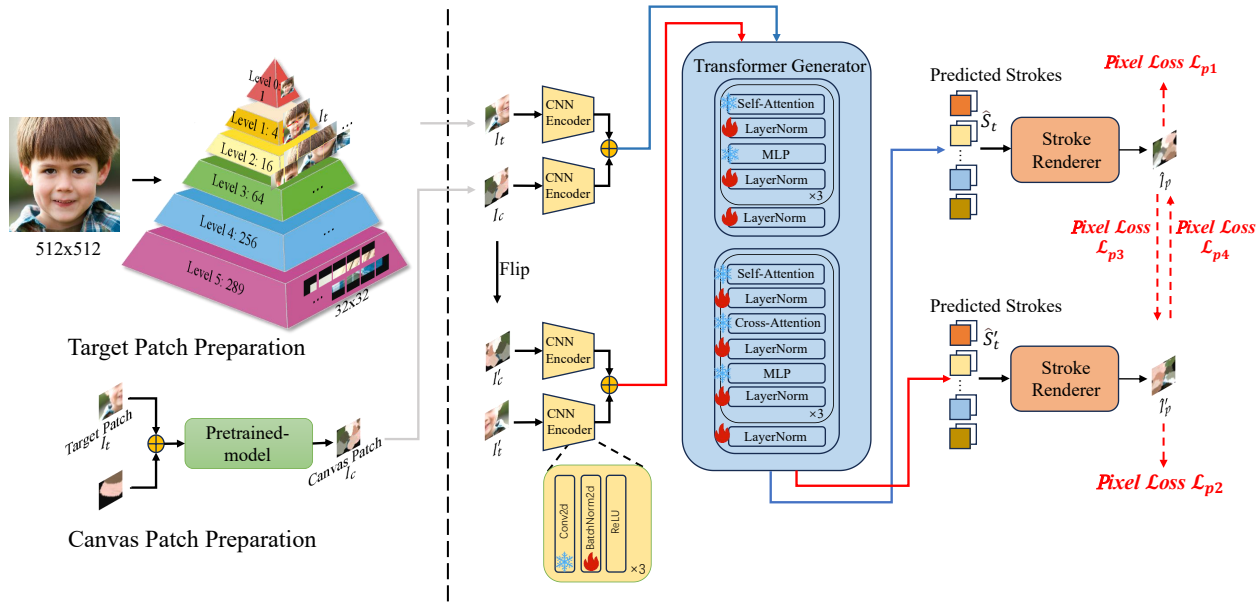

Figure 2: Pipeline overview. Given one single input image of 512 × 512, we split it into different levels according to a pyramid hierarchy as a training set. At each level, the resolution of the image gradually increases until the bottom layer generates a boundary feature map of 544 × 544. At each layer, the feature map is divided into 32 × 32 image patches, which are then input into the Pretrained model for further processing. II. During the Dual-Branch adaptative optimization process, we adopt a freezing strategy for the pretrained model, freezing all other layers except BatchNorm and LayerNorm during the training process to keep the remaining parameters unchanged. The test dataset obtained in the first stage is divided into two parallel processing routes, one of which horizontally flips the image patch and the current patch, and inputs them in batches into the pretrained model. These two parallel circuits use their respective pixel loss calculations to optimize the output results of image blocks, gradually adjusting the parameters of the pretrained model.

Transformer (based on feedforward networks) [23] represent the forefront of advancements in this field. Liu *et al.* [23] introduce the Paint Transformer, which models the neural painting task as a stroke set prediction problem and eliminates the dependency on annotated data through a self-training mechanism. Its core lies in leveraging the self-attention mechanism of the Transformer to capture long-range dependencies within images, generating highly stable painting results by incorporating contextual information. However, this method suffers from issues such as blurred stroke boundaries and granularity mismatches, limiting its application in complex real-world scenarios. Hu *et al.* [13] propose a phase-based reinforcement learning strategy in the Compositional Neural Painter, where a synthesizer network dynamically predicts painting regions and a WGAN-driven [10, 13] stroke renderer generates parameters. While this method addresses boundary artifacts from traditional block-based rendering, the sensitivity of its reinforcement learning agent to out-of-distribution images leads to reduced stability in generated outputs, particularly in heterogeneous lighting scenarios where performance significantly degrades.

Test-Time Adaptation (TTA). Traditional test-time training [31] requires joint optimization of supervised and self-supervised objectives, which further exacerbates the instability of generated quality. Therefore, we shift our focus to TTA, which has two main representatives: 1) hypothesis-transfer-based methods, such as SHOT [20], that achieve target domain adaptation through self-supervised fine-tuning of feature extractors; and 2) entropy minimization based methods, such as Tent [34], that reduce distribution discrepancies by optimizing the statistics of normalization layers. Although these methods perform excellently in classification tasks, their application to generative tasks faces two major limitations: 1) they fail to maintain artistic style consistency, and 2) the parameter update mechanisms are incompatible with the need for decoupling hierarchical features in generative models. Our study is the first to systematically introduce the TTA mechanism into the neural oil painting generation task, proposing a layer normalization progressive adaptation strategy for generative models. This approach resolves the aforementioned limitations while achieving zero-shot domain alignment.

3 DANP: A Domain-Agnostic Neural Painting Framework

The DANP framework pioneers TTA in neural oil painting by introducing a resolution-agnostic domain alignment mechanism. As illustrated in Figure 2, DANP operates in two stages: **(1) Multi-Resolution Pyramid Sampling.** Hierarchical decomposition of input images into pyramidal patches. **(2) Dual-Branch Adaptive Optimization.** Parallel processing of original and mirrored patches

with frozen backbone weights, selectively updating affine parameters of normalization layers. This design achieves source-free unsupervised adaptation while preserving artistic style consistency across resolutions. The primary process is shown in Algorithm 1.

Algorithm 1 Painting Inference Algorithm

Input: Target image T with dimensions $H \times W$; Patch size P; Pre-trained model `net_g`.
Output: Rendered image $\hat{T}_t$ and ordered stroke sequence $\hat{S}$.

Stage I: Multi-Resolution Pyramid Sampling
$K = \max(\arg\min_K \{P \times 2^K \geq \max(H, W)\}, 0)$; #Scale calculation.
$C = blank\_canvas$ and $S^{=}\emptyset$;
for $0 \leq k \leq K$ **do**
  Resize T and C to dimensions $(P \times 2^k, P \times 2^k)$ and partition T and C into uniform patches of size (P, P);
  Store corresponding patches from T and C with differential patches in test dataset as I_t and I_c, then update I_c using pre-trained model `net_g`;
end
Extend T and C via padding to dimensions $(P \times 2^K + P, P \times 2^K + P)$ and store boundary patches in test dataset as I_t and I_c; #Boundary area compensation.
Stage II: Dual-Branch Adaptative Optimization
Extract patches of I_t and I_c from test dataset for fine-tuning the `net_g`;
Generate horizontal mirror images I'_t and I'_c;
Employ CNN encoder and Transformer architecture to predict stroke parameters for dual branches, aggregating all patch strokes as $(\hat{S}_t, \hat{C}_t)$ and $(\hat{S}'_t, \hat{C}'_t)$;
Update canvas: $I_c = I_c + renderer(I_c, \hat{S}_t, \hat{C}_t)$;
$I'_c = I'_c + renderer(I'_c, \hat{S}'_t, \hat{C}'_t)$; # Only high-confidence strokes are rendered.
Derive final stroke collection through $\hat{S} = \hat{S} \cup selected(\hat{S})$ and update canvases I_c and I'_c to obtain generated images $\hat{I}_p$ and $\hat{I}'_p$;
Calculate pixel-wise loss metrics: Update pre-trained model using aggregate loss: $\mathcal{L}_{total} = \alpha \mathcal{L}_{mirror} + \beta \mathcal{L}_{cross}$, yielding fine-tuned model `net_g'`;
Utilize fine-tuned model `net_g'` to predict stroke parameter set $\hat{S}$ and render final image $\hat{T}_t$ approximating target artistic style;
return $\hat{T}_t$ *and* $\hat{S}$

3.1 Multi-Resolution Pyramid Sampling

This study adopts a multi-resolution pyramid sampling method based on a transformer network structure to enhance the diversity of data and promote the model's learning of image generation features. Specifically, we use a recursive method that generates image patches of different sizes from the input image $\mathrm{T} \in \mathbb{R}^{512\times512}$. This process generates patches with hierarchical feature representations:

$$I_t = sampling(\mathrm{T}, \mathrm{P}, \mathrm{K}), \tag{1}$$

where *sampling* represents the recursive sampling process, the layer count K is specifically calculated as shown in Algorithm 1. Ultimately, this process generates the target patch I_t at different hierarchical levels. Each generated patch I_t corresponds to a specific image scale that varies in resolution, and with each increase in layer, the image patch details are progressively refined. Therefore, the hierarchical structure of the transformer model enables the model to learn multi-resolution features of images, improving its representation of artistic tasks.

The corresponding image patches I_t are then processed by the model to generate the final initial canvas I_c that is subsequently used for further fine-tuning. The model utilizes the generated data and learning parameters in the fine-tuning process: $I_c = g(I_t)$, where $g(\cdot)$ denotes the process of generating the initial canvas, using the pre-trained model. In this case, the image patch I_t corresponds to the target image in the transformer architecture, where the generated canvas I_c is resized to 32×32 pixels. The initial canvas created at this stage is crucial for enhancing model performance during the fine-tuning process. By incorporating meaningful brushstroke information, it facilitates more efficient learning for the model and serves as an optimal starting point for subsequent fine-tuning tasks.

3.2 Dual-Branch Adaptative Optimization

Selective Update of Normalization Affine Parameters. Normalization layers, such as Batch Normalization (BatchNorm) [1] and Layer Normalization (LayerNorm), are critical components in deep neural networks. They stabilize training, accelerate convergence, and mitigate issues like vanishing/exploding gradients by normalizing layer inputs. Crucially, both incorporate *learnable affine parameters* (scale γ and shift β) after normalization, allowing the network to preserve or transform the normalized distribution. The core distinction lies in the *axis of normalization*: BatchNorm uses batch+spatial axes, making its statistics dependent on the batch composition. LayerNorm uses the feature axis, making its statistics sample-specific and independent of batch size. This difference impacts their sensitivity to distribution shifts. The normalization operation with affine transformation is defined as:

$$\text{output} = \gamma \cdot \left(\frac{x - \mu}{\sqrt{\sigma^2 + \epsilon}} \right) + \beta, \tag{2}$$

where γ and β are learnable affine parameters (scale and shift), μ and σ are statistical moments (non-learnable), and ϵ is a small constant for numerical stability. During adaptation, only affine parameters (γ, β) are updated via gradient descent while all other parameters (including μ, σ and non-normalization layer weights) remain frozen. Updating γ and β allows the model to efficiently *re-calibrate feature scales and shifts* in response to new data distributions (e.g., artistic styles in image generation), without altering the pre-trained convolutional filters or linear transformations that encode high-level features. Freezing other parameters preserves knowledge from large-scale pre-training and drastically reduces overfitting risk. BatchNorm's affine parameters are particularly sensitive to domain shifts due to their dependence on batch statistics, making their adaptation crucial. LayerNorm's affine parameters offer sample-wise adaptability beneficial for variable-length inputs or small batches. This selective update strikes a balance between adaptability and stability.

Asymmetric Dual-Branch Architecture. In the oil painting generation task, the symmetry and consistency of both local structures and overall layouts of the image are crucial for producing high-quality oil paintings. During the initial experiments, it was observed that a single branch lacked robustness when handling diverse inputs (e.g., different levels and scales of image patches generated in the first stage). As a result, the generated brushstrokes were prone to instability when confronted with complex scenes. To address this, we propose an asymmetric dual-branch architecture,

which introduces horizontal image flipping into the original branch structure. This allows the model to learn feature information from different angles, ensuring that the generated brushstrokes maintain consistency regardless of the direction of the input image. The inclusion of the asymmetric dual-branch architecture essentially increases the input of flipped images, enabling the model to learn more symmetric information and preventing inconsistencies in the generated brushstrokes. Additionally, the asymmetric dual-branch architecture adopts multi-view feature inputs, which implicitly enhance the input data. This design improves the model's stability and flexibility, making it more capable of handling complex scenes during the generation process. The loss function for the asymmetric dual-branch architecture can be expressed as:

$$\mathcal{L}_{mirror} = \mathcal{L}_{p1} + \mathcal{L}_{p2} = \mathcal{L}_{pixel}(\hat{I}_p, I_t) + \mathcal{L}_{pixel}(\hat{I}'_p, I'_t). \quad (3)$$

Dual-Branch Interaction Loss. Despite the inclusion of the asymmetric dual-branch architecture, the use of a single pixel loss during neural image painting still makes it challenging to guide the pretrained model towards the desired optimization direction. This is especially true when generating brushstrokes, as we need to consider not only matching the original image but also maintaining the symmetry and consistency of the generated image. In other words, finer brushstroke optimization details are required. To enhance this refinement capability and ensure that the brushstrokes generated by the model at different levels and perspectives remain consistent with the real image, we have designed a multi-loss strategy.

In addition to the pixel loss between the target and generated images in the original and mirror branches, we introduce two interaction pixel losses between the branches. These loss functions aim to benchmark the original branch and mirror branch outputs against each other, ensuring that the generation results of each branch align with the output of the other branch. This constraint between the brushstrokes generated by different branches further strengthens the model's robustness and consistency during the generation process. The cross-image loss can be formulated as:

$$\mathcal{L}_{p3} = \mathcal{L}_{pixel}(HorizontalFlip(\hat{I}'_p), I_t), \quad (4)$$

$$\mathcal{L}_{p4} = \mathcal{L}_{pixel}(\hat{I}_p, HorizontalFlip(I'_t)), \quad (5)$$

$$\mathcal{L}_{cross} = \mathcal{L}_{p3} + \mathcal{L}_{p4}. \quad (6)$$

By introducing the dual-branch interaction loss, the model is able to learn the comparison between the two branches, further improving the consistency of image quality generated by both branches. Therefore, the final loss can be derived as:

$$\mathcal{L}_{total} = \alpha\mathcal{L}_{mirror} + \beta\mathcal{L}_{cross}, \quad (7)$$

where α and β are weight coefficients that control the balance between two branches.

4 Experiment

We conduct comprehensive experiments on various types of datasets to validate the effectiveness and superiority of the proposed optimization strategy. The experiments are divided into several key sections: dataset description, implementation details of the optimization strategy, ablation studies, performance improvements on images of different resolutions, and comparisons with state-of-the-art methods.

4.1 Dataset Description

To evaluate the proposed method's effectiveness and versatility across various image generation tasks, we conducted experiments on three representative datasets: the FFHQ dataset [17], which contains high-quality facial images; the Landscapes dataset [3], consisting of diverse natural scenery images; and the Wikiart dataset [26], featuring a variety of artistic styles. To ensure the comprehensiveness and representativeness of the experiments, we randomly selected 100 images from each dataset, with each image having a resolution of 512x512. During testing, in order to fully leverage the benefits of the TTA strategy, each image was processed in a pyramid-like hierarchical manner, dividing it into image patches of varying resolutions. Specifically, for each image, 630 pairs of target image patches I_t and canvas image patches I_c were generated, providing ample patch information for further optimization.

4.2 Implementation Details

We utilize a convolutional neural network (CNN) module and a Transformer generator to predict the brushstroke parameters from the input real and canvas images. These predicted parameters are then used in the subsequent rendering stage to generate the final image. When the rendered brushstroke region exceeds 75% coverage, the confidence of the renderer is set to 0 to prevent excessive overlap of brushstrokes, which could lead to unnatural results. Additionally, all hyperparameters for the loss functions are set as $\lambda_{p1} = \lambda_{p2} = 10$, and weighted contributions are applied to balance the influence of each loss term. The AdamW optimizer [24] is employed, with a weight decay parameter of 0.05 to mitigate overfitting. The experiments are conducted on a single Nvidia RTX 4090 GPU with a batch size of 32 and run for a total of 32 epochs. To ensure the model gradually adapts to the new task in the early training phases, a warm-up learning rate strategy [9] is used. Specifically, during the first two epochs, the learning rate linearly increases from the initial value of 1×10^{-6} to the set learning rate of 1×7.5^{-3}, after which it remains constant in the third epoch. Starting from the fourth epoch, a cosine decay strategy is applied, gradually reducing the learning rate, which reaches its final value of 1×5^{-4} by the 32nd epoch, ensuring the stability and convergence of the training process.

4.3 Ablation Studies and Further Discussion

Effect of different components. We thoroughly analyze the contribution of each componentwithin the DANP framework to the overall generation performance and present a quantitative comparison of pixel-level ($\mathcal{L}_{pixel}$) and perceptual-level ($\mathcal{L}_{pcpt}$) reconstruction errors, where lower values indicate higher reconstruction quality. All results were generated at a resolution of 512 × 512 by default, as shown in Table 1 (a). The "w/o" notation indicates the removal of the corresponding components. We observe that removing any component leads to a decrease in the model's stability during fine-tuning, resulting in varying degrees of increase in both pixel and perceptual reconstruction.

Effect of different losses. Table 1 (b) displays the generated results under different branch loss combinations. Starting from the baseline, when the mirror loss is introduced, both ($\mathcal{L}_{pixel}$) and

Table 1: The experiments demonstrate the necessity of all components—selectively updating the affine parameters of normalization layers (BatchNorm, LayerNorm), the geometrically robust dual-branch architecture, and the interaction losses between branches—for the effectiveness of the DANP architecture.

(a) Ablation study on the primary components.

Methods	FFHQ [17]		Landscapes [3]		Wikiart [26]		Average	
	$\mathcal{L}_{pixel} \downarrow$	$\mathcal{L}_{pcpt} \downarrow$	$\mathcal{L}_{pixel} \downarrow$	$\mathcal{L}_{pcpt} \downarrow$	$\mathcal{L}_{pixel} \downarrow$	$\mathcal{L}_{pcpt} \downarrow$	$\mathcal{L}_{pixel} \downarrow$	$\mathcal{L}_{pcpt} \downarrow$
w/o BatchNorm	0.048	0.773	0.060	0.768	0.060	0.841	0.056	0.794
w/o LayerNorm	0.049	0.771	0.061	0.760	0.060	0.834	0.057	0.788
w/o Mirrored Branch	0.051	0.801	0.063	0.782	0.063	0.860	0.059	0.814
Ours	**0.048**	**0.769**	**0.059**	**0.756**	**0.059**	**0.830**	**0.055**	**0.785**

(b) Ablation study on different loss functions

Methods	$\mathcal{L}_{p1}$	$\mathcal{L}_{p2}$	$\mathcal{L}_{p3}$ + $\mathcal{L}_{p4}$	Average	
				$\mathcal{L}_{pixel} \downarrow$	$\mathcal{L}_{pcpt} \downarrow$
Baseline	✓			0.057	0.798
w mirror_loss	✓	✓		0.056	0.786
w cross_loss (**Ours**)	✓	✓	✓	**0.055**	**0.785**

Table 2: Quantitative Analysis on different input resolutions, *i.e.*, 512 and 1024. We observe that the proposed method yields better quality in both resolutions.

Methods	Resolutions	FFHQ [17]		Landscapes [3]		Wikiart [26]		Average		FID↓ [4]
		$\mathcal{L}_{pixel} \downarrow$	$\mathcal{L}_{pcpt} \downarrow$	$\mathcal{L}_{pixel} \downarrow$	$\mathcal{L}_{pcpt} \downarrow$	$\mathcal{L}_{pixel} \downarrow$	$\mathcal{L}_{pcpt} \downarrow$	$\mathcal{L}_{pixel} \downarrow$	$\mathcal{L}_{pcpt} \downarrow$	
Pretrained-model [23]	512×512	0.056	0.870	0.067	0.858	0.064	0.900	0.062	0.876	77.262
DANP		**0.048**	**0.769**	**0.059**	**0.756**	**0.059**	**0.830**	**0.055**	**0.785**	**68.662**
Pretrained-model [23]	1024×1024	0.050	0.723	0.065	0.805	0.061	0.812	0.059	0.780	40.424
DANP		**0.036**	**0.570**	**0.055**	**0.709**	**0.052**	**0.742**	**0.047**	**0.674**	**31.854**

($\mathcal{L}_{pcpt}$) show a decrease. Further incorporating the cross loss results in a reduction of ($\mathcal{L}_{pixel}$) from 0.057 to 0.055, while ($\mathcal{L}_{pcpt}$) decreases from 0.798 to 0.785.The gradual refinement of the loss function design allows the model to generate high-quality images more efficiently, ultimately demonstrating the superiority of the full loss function combination.

Scalability to Different Resolution. To comprehensively evaluate the effectiveness of the proposed neural painting framework, we conduct experiments on images with a resolution of 512×512 and further validated its generalization performance in high-resolution 1024×1024 scenes (as shown in Figure 1). The experiments use a pre-trained model based on Paint Transformer as the baseline. The results demonstrate that the images generated by the optimization strategy significantly outperform those produced by the original pre-trained model at both 512×512 and 1024×1024 resolutions. The brushstroke fineness is improved, with finer details in the generated images and more accurate brushstroke rendering in detailed regions, significantly reducing visible brush traces. The visual quality is enhanced, and image fidelity, color consistency, and boundary coherence are systematically optimized. We also designed a quantitative validation framework, incorporating perceptual loss in addition to pixel loss to establish a dual-metric evaluation system, providing a comprehensive measure of perceptual quality and detail recovery. Table 2 shows that the average performance improvement exceeds 10% at 512×512 resolution, with a further increase of over 20% at 1024×1024 resolution, breaking through the traditional limitation of "high-resolution requiring exponential growth in brushstrokes with diminishing returns." Through the Fréchet Inception Distance (FID) [4] to quantify the distribution differences between the training set (brushstroke domain) and the test set (real-world domain), it is confirmed that the strategy effectively narrows the domain gap.

4.4 Comparison with the state-of-the-arts

Our method is rigorously compared with state-of-the-art neural painting techniques, as shown in Figure 3 and Table 3. To assess its generalization ability, we further apply DANP to the CNP framework (denoted as CNP+DANP). Quantitative results demonstrate that, at a resolution of 512×512, DANP surpasses the baseline methods in both pixel loss and perceptual loss, resulting in a significant improvement in generated image quality. Although CNP, trained on CelebA-HQ [16] and ImageNet [5] using reinforcement learning, performs excellently in portrait generation, it exhibits significant

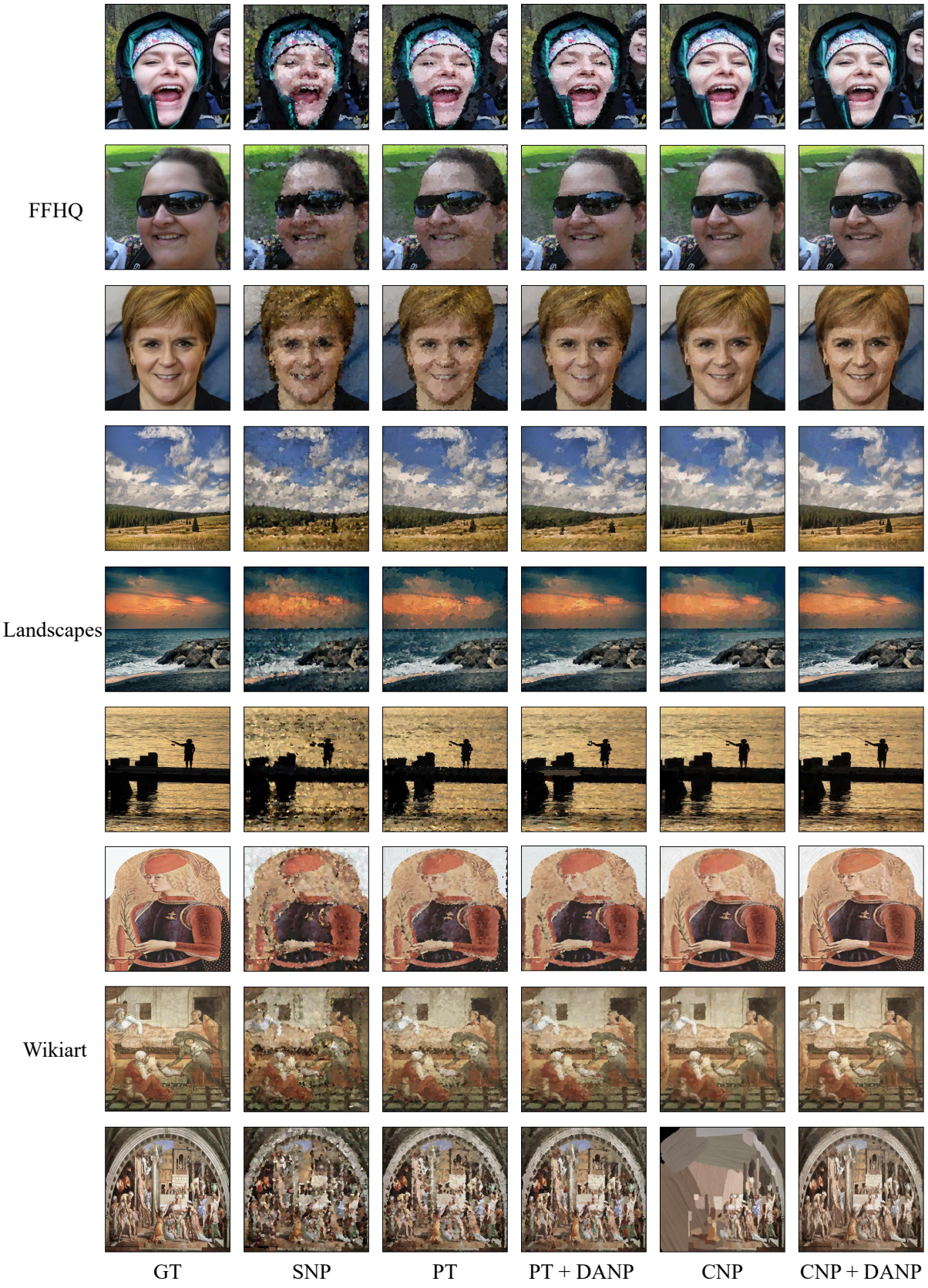

Figure 3: Qualitative comparison with the state-of-the-art methods, including SNP (Stylized Neural Painting [41]), PT (Paint Transformer [23]), and CNP (Composition Neural Painter [13]), on datasets from three domains, *i.e.*, FFHQ, Landscapes, and Wikiart. We observe that the proposed method with adaptation to the target scenario has achieved better visual quality.

Table 3: Quantitative comparison with the state-of-the-art methods. We could observe two points that (1) the proposed method is scalable to different approaches to further improve the output quality (2) CNP with the proposed DANP has achieves better pixel reconstruction, while Paint transformer with ours yeilds better perceptual quality.

Methods	FFHQ [17]		Landscapes [3]		Wikiart [26]		Average	
	$\mathcal{L}_{\text{pixel}} \downarrow$	$\mathcal{L}_{\text{pcpt}} \downarrow$	$\mathcal{L}_{\text{pixel}} \downarrow$	$\mathcal{L}_{\text{pcpt}} \downarrow$	$\mathcal{L}_{\text{pixel}} \downarrow$	$\mathcal{L}_{\text{pcpt}} \downarrow$	$\mathcal{L}_{\text{pixel}} \downarrow$	$\mathcal{L}_{\text{pcpt}} \downarrow$
Stylized Neural Painting [41]	0.070	1.043	0.083	1.098	0.086	1.148	0.079	1.096
Paint Transformer [23]	0.056	0.870	0.067	0.858	0.064	0.900	0.062	0.876
PT+DANP	0.048	**0.769**	0.059	**0.756**	0.059	**0.830**	0.055	**0.785**
Compositional Neural Painter [13]	0.038	0.873	0.048	0.875	0.048	0.908	0.045	0.885
CNP+DANP	**0.036**	0.838	**0.046**	0.843	**0.045**	0.858	**0.042**	0.846

Figure 4: Comparison of generated images with varying numbers of brushstrokes, we observe that DANP achieves high-quality reconstruction even with a low brushstroke count. As the number of brushstrokes increases, DANP consistently outperforms existing method, maintaining a significant advantage.

stability issues in complex scenes, with extreme cases leading to generation failure. In contrast, CNP+DANP not only achieves state-of-the-art (SOTA) performance at the pixel level but also alleviates the instability issues associated with CNP. This demonstrates that DANP serves both as a performance enhancer and a stability stabilizer. Similarly, PT+DANP reaches the optimal performance at the perceptual level, highlighting DANP's ability to amplify the advantages of existing models and its strong generalization capacity.

Inference speed. During inference, we further compare the generated images with varying numbers of brushstrokes, as illustrated in Figure 4. The results show faster convergence and higher reconstruction accuracy, indirectly suggesting that fewer brushstrokes are required to achieve comparable quality, thereby confirming its superior balance between efficiency and accuracy. The proposed framework efficiently selects the salient strokes.

5 Conclusion

This study proposes DANP, a Test-Time Adaptation (TTA)-based Neural Oil Painting method to address the domain gap issue between the real-world test image and the training dataset. In particular, we selectively update the affine parameters of Norm layers, set up mirror branches, and introduce multiple pixel loss functions. While preserving the fidelity of the generated images, the approach enhances brushstroke details, reduces brushstroke artifacts, and further resolves the boundary consistency issue. We conducted extensive comparative experiments, and ablation studies quantitatively demonstrated the necessity of each component of the proposed optimization strategy. Furthermore, the Fréchet Inception Distance (FID) metric confirmed a significant reduction in domain bias. Additionally, we generalized DANP on the CNP framework, and both qualitative and quantitative results show that our method outperforms existing state-of-the-art Neural Painting techniques. Moreover, under the condition of the same number of brushstrokes, our approach is capable of generating more precise and meaningful brushstrokes.

Acknowledgments. We really appreciate the supports from the Macao Science and Technology Development Fund under Grant 0139/2023/RIA2 and 0043/2025/RIA1, and the University of Macau under Grant MYRG-CRG2024-00037-FST-ICI and Nanjing Municipal Science and Technology Bureau 202401035, and University of Macau SRG2024-00002-FST.

References

[1] John Bronskill, Jonathan Gordon, James Requeima, Sebastian Nowozin, and Richard Turner. 2020. Tasknorm: Rethinking batch normalization for meta-learning. In *International Conference on Machine Learning*. PMLR, 1153–1164.

[2] Silvia Bucci, Antonio D'Innocente, Yujun Liao, Fabio M Carlucci, Barbara Caputo, and Tatiana Tommasi. 2021. Self-supervised learning across domains. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 9 (2021), 5516–5528.

[3] Yang Chen, Yu-Kun Lai, and Yong-Jin Liu. 2018. Cartoongan: Generative adversarial networks for photo cartoonization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 9465–9474.

[4] Min Jin Chong and David Forsyth. 2020. Effectively unbiased fid and inception score and where to find them. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 6070–6079.

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255.

[6] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. 2014. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*. PMLR, 647–655.

[7] Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*. PMLR, 1180–1189.

[8] Xiang Gao, Yingjie Tian, and Zhiquan Qi. 2020. RPD-GAN: Learning to draw realistic paintings with generative adversarial network. *IEEE Transactions on Image Processing* 29 (2020), 8706–8720.

[9] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. 2017. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677* (2017).

[10] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. 2017. Improved training of wasserstein gans. *Advances in neural information processing systems* 30 (2017).

[11] Paul Haeberli. 1990. Paint by numbers: Abstract image representations. In *Proceedings of the 17th annual conference on Computer graphics and interactive techniques*. 207–214.

[12] Aaron Hertzmann. 2003. A survey of stroke-based rendering. In *IEEE Transactions on Visualization and Computer Graphics*. Institute of Electrical and Electronics Engineers.

[13] Teng Hu, Ran Yi, Haokun Zhu, Liang Liu, Jinlong Peng, Yabiao Wang, Chengjie Wang, and Lizhuang Ma. 2023. Stroke-based neural painting and stylization with dynamically predicted painting region. In *Proceedings of the 31st ACM International Conference on Multimedia*. 7470–7480.

[14] Zhewei Huang, Wen Heng, and Shuchang Zhou. 2019. Learning to paint with model-based deep reinforcement learning. In *Proceedings of the IEEE/CVF international conference on computer vision*. 8709–8718.

[15] Hong Jia, Young Kwon, Alessio Orsino, Ting Dang, Domenico Talia, and Cecilia Mascolo. 2024. TinyTTA: Efficient Test-time Adaptation via Early-exit Ensembles on Edge Devices. *Advances in Neural Information Processing Systems* 37 (2024), 43274–43299.

[16] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. 2017. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196* (2017).

[17] Tero Karras, Samuli Laine, and Timo Aila. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4401–4410.

[18] Dmytro Kotovenko, Matthias Wright, Arthur Heimbrecht, and Bjorn Ommer. 2021. Rethinking style transfer: From pixels to parameterized brushstrokes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12196–12205.

[19] Jian Liang, Ran He, and Tieniu Tan. 2025. A comprehensive survey on test-time adaptation under distribution shifts. *International Journal of Computer Vision* 133, 1 (2025), 31–64.

[20] Jian Liang, Dapeng Hu, Yunbo Wang, Ran He, and Jiashi Feng. 2021. Source data-absent unsupervised domain adaptation through hypothesis transfer and labeling transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 11 (2021), 8602–8617.

[21] Peter Litwinowicz. 1997. Processing images and video for an impressionist effect. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*. 407–414.

[22] Lingyu Liu, Yaxiong Wang, Li Zhu, and Zhedong Zheng. 2025. Every Painting Awakened: A Training-free Framework for Painting-to-Animation Generation. *arXiv preprint arXiv:2503.23736* (2025).

[23] Songhua Liu, Tianwei Lin, Dongliang He, Fu Li, Ruifeng Deng, Xin Li, Errui Ding, and Hao Wang. 2021. Paint transformer: Feed forward neural painting with stroke prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*. 6598–6607.

[24] Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101* (2017).

[25] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. 2022. Efficient test-time model adaptation without forgetting. In *International conference on machine learning*. PMLR, 16888–16905.

[26] Fred Phillips and Brandy Mackintosh. 2011. Wiki art gallery, inc.: A case for critical thinking. *Issues in Accounting Education* 26, 3 (2011), 593–608.

[27] Gal Shalev, Gabi Shalev, and Joseph Keshet. 2022. A baseline for detecting out-of-distribution examples in image captioning. In *Proceedings of the 30th ACM International Conference on Multimedia*. 4175–4184.

[28] Jaskirat Singh, Cameron Smith, Jose Echevarria, and Liang Zheng. 2021. Intelli-paint: Towards developing human-like painting agents. *arXiv preprint arXiv:2112.08930* (2021).

[29] Jaskirat Singh, Cameron Smith, Jose Echevarria, and Liang Zheng. 2022. Intelli-Paint: Towards developing more human-intelligible painting agents. In *European conference on computer vision*. Springer, 685–701.

[30] Guoxian Song, Linjie Luo, Jing Liu, Wan-Chun Ma, Chunpong Lai, Chuanxia Zheng, and Tat-Jen Cham. 2021. Agilegan: stylizing portraits by inversion-consistent transfer learning. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 1–13.

[31] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei A Efros, and Moritz Hardt. 2019. Test-time training for out-of-distribution generalization. *Proceedings of the National Academy of Sciences* 116, 13 (2019), 6234–6242. doi:10.1073/pnas.1815990116

[32] Zhengmi Tang, Tomo Miyazaki, Yoshihiro Sugaya, and Shinichiro Omachi. 2021. Stroke-based scene text erasing using synthetic data for training. *IEEE Transactions on Image Processing* 30 (2021), 9306–9320.

[33] Zhengyan Tong, Xiaohang Wang, Shengchao Yuan, Xuanhong Chen, Junjie Wang, and Xiangzhong Fang. 2022. Im2oil: Stroke-based oil painting rendering with linearly controllable fineness via adaptive sampling. In *Proceedings of the 30th ACM international conference on multimedia*. 1035–1046.

[34] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. 2020. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726* (2020).

[35] Zunfu Wang, Fang Liu, Zhixiong Liu, Changjuan Ran, and Mohan Zhang. 2024. Intelligent-paint: a Chinese painting process generation method based on vision transformer. *Multimedia Systems* 30, 2 (2024), 112.

[36] Yanan Wu, Zhixiang Chi, Yang Wang, Konstantinos N Plataniotis, and Songhe Feng. 2024. Test-time domain adaptation by learning domain-aware batch normalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 15961–15969.

[37] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks? *Advances in neural information processing systems* 27 (2014).

[38] Shaozu Yuan, Ruixue Liu, Meng Chen, Baoyang Chen, Zhijie Qiu, and Xiaodong He. 2021. Learning to compose stylistic calligraphy artwork with emotions. In *Proceedings of the 29th ACM International Conference on Multimedia*. 3701–3709.

[39] Zhedong Zheng and Yi Yang. 2022. Adaptive boosting for domain adaptation: Toward robust predictions in scene segmentation. *IEEE Transactions on Image Processing* 31 (2022), 5371–5382.

[40] Yifei Zhou, Juntao Ren, Fengyu Li, Ramin Zabih, and Ser Nam Lim. 2023. Test-time distribution normalization for contrastively learned visual-language models. *Advances in Neural Information Processing Systems* 36 (2023), 47105–47123.

[41] Zhengxia Zou, Tianyang Shi, Shuang Qiu, Yi Yuan, and Zhenwei Shi. 2021. Stylized neural painting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 15689–15698.