INFERENCE-TIME TRAJECTORY CORRECTION FOR STRUCTURE-PRESERVING MANGA IMAGE EDITING

Anonymous authors
Paper under double-blind review

ABSTRACT

We present a lightweight, training-free trajectory correction method that adapts a pretrained image editing model to each input manga image using only the input itself. Despite recent progress in pretrained image editing, such models often underperform on manga because they are trained predominantly on natural-image data, while re-training or fine-tuning them on manga is costly and raises copyright concerns. Many manga image editing tasks encountered in practice are structure-preserving, requiring local details to be modified while the input's global composition is retained. To support this common editing setting, our method corrects the early editing trajectory by anchoring it to an empty-prompt reconstruction trajectory. Experiments on text removal show consistent improvements over direct inference with both Qwen Image Edit 2509 and FLUX.1 Kontext, and an additional application to screentone synthesis shows that the method also better preserves the input composition. With FLUX.1 Kontext on an RTX A6000, the method incurs 11% runtime overhead and 0.1% peak-memory overhead.

1 INTRODUCTION

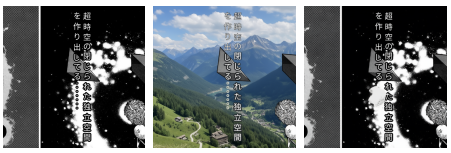
Manga image editing is an important task in visual content creation. Manga, a form of Japanese comics, has attracted a broad global audience, creating demand for a variety of image editing techniques. For instance, when manga is localized into other languages, text embedded in the image must be removed seamlessly as a preprocessing step. Moreover, a method that directly transforms line art into a completed manga image with screentones would provide practical assistance to creators.

Recently, the computer vision community has seen rapid progress in image editing methods built on large pretrained image generation models (Huang et al., 2025). In particular, methods based on Rectified Flow (Liu et al., 2023; Lipman et al., 2023), which enable high-quality image generation, have become the dominant paradigm (Kulikov et al., 2025; Deng et al., 2025; Rout et al., 2025; Wang et al., 2025; Patel et al., 2025; Hu et al., 2024). These methods perform editing by re-generating an image that follows a target prompt from an input image associated with a source prompt using a pretrained image generation model. More recently, models such as Qwen Image Edit and FLUX.1 Kontext, which are trained explicitly for image editing (hereafter, *image editing models*), have emerged (Wu et al., 2025; Labs et al., 2025; Liu et al., 2025; Team et al., 2025). These pretrained image editing models can be driven by direct editing instructions (e.g., "Remove text." for removing text from an image), enabling more intuitive and flexible editing than methods based on generic image generation models.

However, these pretrained image editing models often do not work well when directly applied to manga image editing. For example, when we feed the manga image in Fig. 1a to the model and provide an "empty prompt" that specifies no edit, the model still transforms the image into a natural-image-like result as shown in Fig. 1b. This failure to preserve the input even when no edit is requested suggests that the same domain bias can distort the global composition during local manga edits. A plausible reason is that the data used to train these image editing models consist mostly of natural images, while manga images are relatively scarce.

A straightforward remedy would be to collect manga images and fine-tune an image editing model, but this is often impractical. Image editing models such as Qwen Image Edit and FLUX.1 Kontext are large and require substantial computational resources for training. Even if efficient adaptation

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



(a) Input (b) Baseline (c) Ours

Figure 1: Editing results with an empty prompt. ©Kato Masaki



(a) Text removal (b) Screentone synthesis

Figure 2: Examples of manga image editing. In each pair, the left image is the input and the right image is the ground truth. ©Kato Masaki ©Yabuno Tenya, Watanabe Tatsuya

methods such as LoRA (Hu et al., 2022) are used, collecting manga images and using them for training still raises copyright concerns.

To address this issue, we propose an input-specific correction of the inference trajectory that requires neither parameter updates nor external manga data. Our approach builds on three observations. First, Rectified Flow-based image editing models, including Qwen Image Edit and FLUX.1 Kontext, synthesize an edited image from noise while conditioning on the input image, rather than directly modifying its pixels. Their editing behavior can therefore be adjusted by intervening in the generation trajectory at inference time. Second, many practical manga editing tasks, including text removal and screentone synthesis, are structure-preserving: the output should retain the input's global composition while modifying only local details (Fig. 2). Here, we use *global composition* to refer collectively to the spatial layout and the structural line art that defines characters, objects, and panels; local details include elements such as text and screentone patterns. Third, diffusion/flow models determine global structure mainly in the early timesteps and refine details in later timesteps (Qian et al., 2024; Wang & Vastola, 2023; Ma et al., 2025). Together, these observations motivate correcting only the first few steps of the editing trajectory, where global composition is primarily determined. Specifically, at each corrected step, we use the model's empty-prompt prediction to construct a correction target whose predicted endpoint is the input image and anchor the editing trajectory toward this target. The resulting correction is lightweight and practical: it introduces no trainable module, leaves the pretrained architecture unchanged, and can be incorporated into existing Rectified Flow-based image editing models through a sampler-level modification.

We evaluate the proposed method mainly on text removal, a practically important task in manga localization for which exact pairs of input and ground-truth images can be constructed for quantitative evaluation. Because the method assumes only a Rectified Flow-based editing process and does not depend on a particular network architecture, we test it across several Qwen and FLUX backbones. On text removal, the correction consistently improves direct inference with both Qwen Image Edit 2509 and FLUX.1 Kontext, and it applies to the other backbones without model-specific training. We further apply the same correction to screentone synthesis, which adds local visual patterns to line art rather than removing content. The results in the supplementary material show that the correction also better preserves the input composition in this task.

The correction is also lightweight: the edit- and empty-prompt predictions are computed in a single batched model call at each corrected step, avoiding an additional sequential network pass. With FLUX.1 Kontext, the measured overhead is 11% in runtime and 0.1% in peak memory (Sec. 5.5).

Finally, the input anchor makes the method particularly well suited to local edits that preserve global composition. One current limitation is that the correction is not designed for edits requiring substantial changes in shape, pose, layout, or semantics. Extending the method with an adaptive anchoring mechanism for such edits is an important direction for future work.

Our contributions are summarized as follows:

- To the best of our knowledge, this is the first inference-time adaptation method for instruction-based image editing models, which take an input image and an edit instruction.
- We introduce a lightweight, training-free trajectory correction that uses empty-prompt reconstruction as an input-specific anchor for preserving global composition, without training or architectural changes.

- Experiments on text removal show consistent improvements over direct inference on two image editors with small practical overhead, and an additional application to screentone synthesis shows better composition preservation.

2 Related Work

2.1 Test-Time Adaptation

Test-time adaptation, which aims to handle distribution shifts between training and inference data, has been studied extensively in image classification (Liang et al., 2025). A representative line of work updates a subset of model parameters, such as those in batch normalization layers, at inference time using the predicted class probabilities (Wang et al., 2021; Niu et al., 2022). More recently, methods have also been proposed to update either the input image or model parameters at inference time by leveraging feedback or guidance from image generation models (Prabhudesai et al., 2023; Tsai et al., 2024; Guo et al., 2025). However, these methods typically suffer from one or more of the following limitations: architectural constraints, task specificity to image classification, or substantial additional computational cost at inference time.

Chen et al. (2025) proposed an efficient method for large pretrained segmentation models that updates image embeddings (latent representations). However, their method, like the classification-oriented approaches above, relies on classification or segmentation outputs and objectives that are unavailable in an image editing model; it therefore cannot be directly applied to our setting.

In contrast to these prior works, we present the first inference-time adaptation method for recently emerged instruction-based image editing models. Test-time scaling methods for such models (Qu et al., 2026; Yu et al., 2026) sample multiple outputs and select among them, whereas our method adapts the generation process itself to each input. Instead of a classifier- or segmenter-based adaptation objective, our method derives a correction directly from the reconstruction behavior of the image editing model.

2.2 Inversion and Trajectory-Based Editing

Recent training-free methods edit images with flow-based generation models by modifying their inversion, sampling dynamics, or intermediate representations. RF-Inversion (Rout et al., 2025) derives an inversion vector field via dynamic optimal control, FireFlow (Deng et al., 2025) accelerates inversion by reusing the preceding step's midpoint velocity, and RF-Solver (Wang et al., 2025) reduces numerical errors in inversion, with RF-Edit preserving source structure via inversion features. Other methods include Hu et al. (2024), who edit through a controllable u-space and localized prompts, and FlowEdit (Kulikov et al., 2025), which constructs an inversion-free ODE between source and target distributions. Stable Flow (Avrahami et al., 2025) injects attention features into vital layers, ReFlex (Kim et al., 2025) uses mid-step feature extraction and attention adaptation, and Consistent-Inversion (Lu et al., 2026) corrects early target denoising steps with a reverse-consistency signal toward the source inversion trajectory. The most closely related is FlowChef (Patel et al., 2025), which steers the vector field with an endpoint objective (Sec. 4.3). Unlike these methods, we correct the early trajectory of an image editing model that already takes an input image and an edit instruction. To preserve global composition during local editing, we anchor this trajectory to the same model's empty-prompt prediction. This correction requires neither inversion, an external objective, nor source/target prompts. Because the edit-prompt velocity is retained for the Euler update, the response to the edit instruction is also preserved (Proposition 2). We compare with representative inversion and trajectory-based methods in Sec. 5.4.

2.3 Manga Image Editing

A wide range of tasks have been studied in manga image editing. Representative examples include automatic colorization (Golyadkin et al., 2025; Hensman & Aizawa, 2017; Furusawa et al., 2017; Shimizu et al., 2021; Qu et al., 2006; Kataoka et al., 2017; Sato et al., 2014), text removal (Ko & Cho, 2020; Xie et al., 2021), screentone synthesis (Tsubota et al., 2019; Lin et al., 2024), and retargeting (Matsui et al., 2011; Xie et al., 2025). Unlike these task-specific approaches, our method adapts a pretrained image editing model to each input at inference time without training a dedicated

model for the target task. It therefore avoids collecting a task-specific manga training set and can change the requested local edit through the instruction prompt. Our experiments evaluate this task-flexible adaptation on text removal and screentone synthesis, two complementary tasks for which quantitative evaluation can be constructed using manga with documented research-use permission.

3 METHOD

3.1 PRELIMINARIES: GENERATION PROCESS OF IMAGE EDITING MODELS

Rectified Flow-based image editing models such as Qwen Image Edit and FLUX.1 Kontext do not edit the input image directly. Instead, conditioned on the input image, they generate an edited image from noise. This generation process can be written as the following ordinary differential equation:

$$d\boldsymbol{Z}_t = \boldsymbol{v}_\theta(\boldsymbol{Z}_t, t, c, \boldsymbol{X}_{\text{in}})dt, \tag{1}$$

where $t \in [0, 1]$ denotes time, c is the editing prompt, $\boldsymbol{Z}_0 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ is Gaussian noise, $\boldsymbol{Z}_1 = \boldsymbol{X}_{\text{edit}}$ is the edited image, and $\boldsymbol{v}_\theta$ is the velocity predicted by the pretrained image editing model. In practice, t is discretized into N timesteps $t = (t_0, \cdots, t_N)$, and generation uses an Euler solver:

$$\boldsymbol{Z}_{t_{i+1}} = \boldsymbol{Z}_{t_i} + (t_{i+1} - t_i)\boldsymbol{v}_\theta(\boldsymbol{Z}_{t_i}, t_i, c, \boldsymbol{X}_{\text{in}}). \tag{2}$$

Starting from $t_0 = 0$ and ending at $t_N = 1$, we iterate the above update for $i \in \{0, \cdots, N-1\}$ to obtain the edited image $\boldsymbol{Z}_1 (= \boldsymbol{X}_{\text{edit}})$. These models operate in the latent space of a VAE; we therefore write $\boldsymbol{X}_{\text{in}}$ for the input-image latent and obtain the edited image by decoding $\boldsymbol{Z}_1$.

3.2 CORRECTING THE GENERATION TRAJECTORY

The key difficulty is that when $\boldsymbol{X}_{\text{in}}$ lies outside the training distribution, as is often the case for manga images, the predicted velocity $\boldsymbol{v}_\theta(\boldsymbol{Z}_{t_i}, t_i, c, \boldsymbol{X}_{\text{in}})$ becomes inaccurate, leading to trajectories and outputs that are unsuitable for manga images. Our method addresses this issue by correcting the generation trajectory.

Algorithm 1 Inference-time trajectory correction

Require: Pretrained Rectified Flow model $\boldsymbol{v}_\theta$, input image $\boldsymbol{X}_{\text{in}}$, noise $\boldsymbol{Z}_0$, and edit prompt c

for $i \in \{0, \cdots, N-1\}$ **do**
  $\boldsymbol{v} \leftarrow \boldsymbol{v}_\theta(\boldsymbol{Z}_{t_i}, t_i, c, \boldsymbol{X}_{\text{in}})$
  if $i < M$ **then**
    $\boldsymbol{u} \leftarrow \boldsymbol{v}_\theta(\boldsymbol{Z}_{t_i}, t_i, \emptyset, \boldsymbol{X}_{\text{in}})$
    $\boldsymbol{Z}^*_{t_i} \leftarrow \boldsymbol{X}_{\text{in}} - (t_N - t_i)\boldsymbol{u}$
    $\boldsymbol{Z}_{t_i} \leftarrow (1-\alpha)\boldsymbol{Z}_{t_i} + \alpha\boldsymbol{Z}^*_{t_i}$
  $\boldsymbol{Z}_{t_{i+1}} \leftarrow \boldsymbol{Z}_{t_i} + (t_{i+1} - t_i)\boldsymbol{v}$
return $\boldsymbol{Z}_{t_N}$

A key property of manga editing tasks such as text removal and screentone synthesis is that the global composition of the input image $\boldsymbol{X}_{\text{in}}$ and the edited image $\boldsymbol{X}_{\text{edit}}$ is expected to remain largely unchanged. In other words, the edited output should preserve the input's global composition while changing only local details. Given the observation that diffusion/flow models generate global structure in early timesteps and fine details in later timesteps (Qian et al., 2024; Wang & Vastola, 2023; Ma et al., 2025), it is desirable for the trajectory of the edited image to stay close to that of the input image during the first $M (< N)$ steps, $t = (t_0, \cdots, t_M)$. We therefore compute a trajectory that reconstructs the input image under an empty prompt and use it to correct the trajectory of the edited image.

More specifically, at timestep t_i, let $\boldsymbol{u}_i := \boldsymbol{v}_\theta(\boldsymbol{Z}_{t_i}, t_i, \emptyset, \boldsymbol{X}_{\text{in}})$ be the empty-prompt velocity evaluated at the current state. We freeze this velocity over the remaining interval, i.e., take a single Euler step from t_i to t_N, and define the resulting frozen-velocity endpoint estimate for a candidate state $\boldsymbol{Z}$ as

$$\hat{\boldsymbol{Z}}_1(\boldsymbol{Z}) = \boldsymbol{Z} + (t_N - t_i)\boldsymbol{u}_i. \tag{3}$$

We then define the correction target by minimizing the discrepancy between this estimate and $\boldsymbol{X}_{\text{in}}$:

$$\boldsymbol{Z}^*_{t_i} = \arg\min_{\boldsymbol{Z}} \left\| \boldsymbol{X}_{\text{in}} - \hat{\boldsymbol{Z}}_1(\boldsymbol{Z}) \right\|_2^2 \tag{4}$$

$$= \boldsymbol{X}_{\text{in}} - (t_N - t_i)\boldsymbol{u}_i. \tag{5}$$

Using this result, we correct the original trajectory $\boldsymbol{Z}_{t_i}$ as follows:

$$\boldsymbol{Z}_{t_i} \leftarrow (1-\alpha)\boldsymbol{Z}_{t_i} + \alpha\boldsymbol{Z}^*_{t_i}, \tag{6}$$

where α is a hyperparameter that controls the correction strength.

3.3 Algorithm and Computational Cost

The full procedure is summarized in Algorithm 1. The red lines indicate the additional computation introduced by our method. For the first M steps, the prediction with the edit prompt c (line 2) and that with the empty prompt $\emptyset$ (line 4) can be executed in one batched model call by concatenating the two conditions. This avoids an additional *sequential* model pass, but still increases the amount of computation and the minibatch memory footprint. The interpolation in lines 5 and 6 adds only simple matrix operations. Both predictions are evaluated at the uncorrected state $\boldsymbol{Z}_{t_i}$, and the edit-prompt velocity $\boldsymbol{v}$ is then used for the Euler update from the corrected state (line 7). This lightweight correction is easy to add to an existing sampler and requires neither modification nor retraining of the underlying model. We report runtime and peak memory for FLUX.1 Kontext on an A6000 GPU and runtime for Qwen Image Edit 2509 on an H200 GPU in Sec. 5.5.

4 Mathematical Interpretation

This section analyzes the proposed correction while holding the velocity predictions computed at timestep t_i fixed, matching how Algorithm 1 constructs and applies the correction target. Sec. 4.1 characterizes the local effect of the interpolation, Sec. 4.2 explains the choice of the empty prompt, and Sec. 4.3 clarifies the mathematical relationship to FlowChef.

4.1 Local Effect of the Interpolation

This subsection shows that the interpolation correction reduces the discrepancy between the input image and the endpoint predicted from the empty-prompt velocity. At timestep t_i, define the empty-prompt velocity $\boldsymbol{u}_i := \boldsymbol{v}_\theta(\boldsymbol{Z}_{t_i}, t_i, \emptyset, \boldsymbol{X}_{\text{in}})$ and the remaining time to the endpoint $\Delta_i := t_N - t_i$. Holding $\boldsymbol{u}_i$ fixed, a candidate current state $\boldsymbol{Z}$ predicts the endpoint $\boldsymbol{Z} + \Delta_i \boldsymbol{u}_i$. The correction target $\boldsymbol{Z}^*_{t_i} = \boldsymbol{X}_{\text{in}} - \Delta_i \boldsymbol{u}_i$ is therefore the unique state whose predicted empty-prompt endpoint equals the input image. For the current state, define the endpoint reconstruction residual as $\boldsymbol{r}_i := \boldsymbol{Z}_{t_i} + \Delta_i \boldsymbol{u}_i - \boldsymbol{X}_{\text{in}}$. Equivalently, the target is $\boldsymbol{Z}^*_{t_i} = \boldsymbol{Z}_{t_i} - \boldsymbol{r}_i$: it removes the entire frozen-velocity residual. Algorithm 1 moves only a fraction α toward this target, $\boldsymbol{Z}^+_{t_i} := (1-\alpha)\boldsymbol{Z}_{t_i} + \alpha \boldsymbol{Z}^*_{t_i}$.

Proposition 1 (Local reconstruction contraction under a frozen velocity). Under the frozen velocity $\boldsymbol{u}_i$, the corrected state is $\boldsymbol{Z}^+_{t_i} = \boldsymbol{Z}_{t_i} - \alpha \boldsymbol{r}_i$, and the residual after correction satisfies

$$\boldsymbol{r}_i^+ := \boldsymbol{Z}^+_{t_i} + \Delta_i \boldsymbol{u}_i - \boldsymbol{X}_{\text{in}} = (1-\alpha)\boldsymbol{r}_i. \tag{7}$$

Consequently,

$$\|\boldsymbol{r}_i^+\|_2 = |1-\alpha|\,\|\boldsymbol{r}_i\|_2, \qquad \|\boldsymbol{r}_i^+\|_2^2 = (1-\alpha)^2 \|\boldsymbol{r}_i\|_2^2. \tag{8}$$

Proof. Substituting $\boldsymbol{Z}^*_{t_i} = \boldsymbol{Z}_{t_i} - \boldsymbol{r}_i$ into the interpolation update gives $\boldsymbol{Z}^+_{t_i} = \boldsymbol{Z}_{t_i} - \alpha \boldsymbol{r}_i$. Adding $\Delta_i \boldsymbol{u}_i - \boldsymbol{X}_{\text{in}}$ to both sides yields $\boldsymbol{r}_i^+ = (1-\alpha)\boldsymbol{r}_i$, from which the norm identities follow. □

Proposition 1 makes the intended immediate effect of each correction step explicit: for $0 < \alpha < 1$, the empty-prompt endpoint estimate at the current step moves strictly toward the input, and $\alpha = 1$ eliminates the frozen-velocity residual. This result requires neither a network Jacobian nor a Lipschitz assumption. Because later steps use newly predicted velocities, we evaluate the cumulative effect on the sampling trajectory and the final output empirically in Sec. 5.4.

4.2 Why the Empty Prompt Helps

To clarify why the correction target is constructed with the empty prompt, we compare the proposed correction with an alternative variant that uses the edit prompt c instead. Both variants use the edit-prompt velocity for the actual Euler update; they differ only in the prompt used to construct the correction target. Let $h_i := t_{i+1} - t_i$ be the Euler step size. Let the edit-prompt velocity and its difference from the empty-prompt velocity be $\boldsymbol{v}_i^{(c)} := \boldsymbol{v}_\theta(\boldsymbol{Z}_{t_i}, t_i, c, \boldsymbol{X}_{\text{in}})$ and $\boldsymbol{\delta}_i := \boldsymbol{v}_i^{(c)} - \boldsymbol{u}_i$, so that $\boldsymbol{v}_i^{(c)} = \boldsymbol{u}_i + \boldsymbol{\delta}_i$.

Proposition 2 (Effect of the prompt used for target construction). The proposed empty-prompt target and the alternative edit-prompt target are, respectively,

$$\boldsymbol{Z}^{*,(\emptyset)}_{t_i} = \boldsymbol{X}_{\text{in}} - \Delta_i \boldsymbol{u}_i, \qquad \boldsymbol{Z}^{*,(c)}_{t_i} = \boldsymbol{X}_{\text{in}} - \Delta_i(\boldsymbol{u}_i + \boldsymbol{\delta}_i). \tag{9}$$

For either choice $p \in \{\emptyset, c\}$, one correction step followed by the Euler update is

$$\boldsymbol{Z}_{t_{i+1}}^{(p)} = (1-\alpha)\boldsymbol{Z}_{t_i} + \alpha \boldsymbol{Z}_{t_i}^{*,(p)} + h_i(\boldsymbol{u}_i + \boldsymbol{\delta}_i). \tag{10}$$

The target $\boldsymbol{Z}_{t_i}^{*,(p)}$ is the only term in this update that differs between the two variants. Substituting the two targets yields

$$\begin{aligned} \boldsymbol{Z}_{t_{i+1}}^{(\emptyset)} &= (1-\alpha)\boldsymbol{Z}_{t_i} + \alpha \boldsymbol{X}_{\text{in}} + (h_i - \alpha\Delta_i)\boldsymbol{u}_i + h_i\boldsymbol{\delta}_i, \\ \boldsymbol{Z}_{t_{i+1}}^{(c)} &= (1-\alpha)\boldsymbol{Z}_{t_i} + \alpha \boldsymbol{X}_{\text{in}} + (h_i - \alpha\Delta_i)\boldsymbol{u}_i + (h_i - \alpha\Delta_i)\boldsymbol{\delta}_i. \end{aligned} \tag{11}$$

Their difference is therefore

$$\boldsymbol{Z}_{t_{i+1}}^{(c)} - \boldsymbol{Z}_{t_{i+1}}^{(\emptyset)} = -\alpha\Delta_i\boldsymbol{\delta}_i. \tag{12}$$

Proof. Substituting each target into the correction-plus-Euler update above gives the two stated expressions. Subtracting them gives the final identity. □

The first three terms in the two next-state expressions are identical; only the coefficient of $\boldsymbol{\delta}_i$ differs. Because $\boldsymbol{\delta}_i$ is the change in velocity caused by replacing the empty prompt with the edit prompt, $h_i\boldsymbol{\delta}_i$ is the part added by the original Euler step in response to the edit instruction. The proposed empty-prompt target keeps its coefficient at h_i, so the proposed correction does not implicitly rescale this response. In contrast, the edit-prompt target introduces $-\alpha\Delta_i\boldsymbol{\delta}_i$ through the correction, changing the coefficient from h_i to $h_i - \alpha\Delta_i$. The coefficient is reduced, becomes zero when $\alpha\Delta_i = h_i$, and becomes negative when $\alpha\Delta_i > h_i$. This interference is most relevant in the early steps, where Δ_i is largest and the proposed correction is applied. Supplementary Sec. B.1 empirically confirms this interference: constructing the target with the edit prompt pulls the output back toward the input and lowers text-region PSNR below the baseline, whereas the empty prompt improves it.

4.3 Connection to FlowChef

To clarify the mathematical relationship to FlowChef, we first identify the update form the two methods share and then clarify how the two methods construct the steering term differently.

In our notation, the FlowChef update (Patel et al., 2025) after gradient skipping is

$$\boldsymbol{Z}_{t_{i+1}} = \boldsymbol{Z}_{t_i} + \underbrace{h_i \boldsymbol{v}_\theta(\boldsymbol{Z}_{t_i}, t_i)}_{\text{base generation step}} - \underbrace{s\nabla_{\hat{\boldsymbol{Z}}_1} L}_{\text{endpoint-objective steering}}, \tag{13}$$

where L is an objective on the predicted endpoint $\hat{\boldsymbol{Z}}_1$ and s is its steering scale. To write the proposed correction in this form, define the endpoint reconstruction loss $\tilde{L}_i(\boldsymbol{Z}) := \|\boldsymbol{Z} + \Delta_i\boldsymbol{u}_i - \boldsymbol{X}_{\text{in}}\|_2^2$. As $\nabla_{\boldsymbol{Z}}\tilde{L}_i(\boldsymbol{Z}_{t_i}) = 2\boldsymbol{r}_i$, the interpolation in Proposition 1 is the exact gradient step

$$\boldsymbol{Z}_{t_i}^{+} = \boldsymbol{Z}_{t_i} - \frac{\alpha}{2}\nabla_{\boldsymbol{Z}}\tilde{L}_i(\boldsymbol{Z}_{t_i}). \tag{14}$$

Combining this correction with the Euler step, the first M steps of the proposed sampler become

$$\boldsymbol{Z}_{t_{i+1}} = \boldsymbol{Z}_{t_i} + \underbrace{h_i \boldsymbol{v}_\theta(\boldsymbol{Z}_{t_i}, t_i, c, \boldsymbol{X}_{\text{in}})}_{\text{base editing step}} - \underbrace{\frac{\alpha}{2}\nabla_{\boldsymbol{Z}}\tilde{L}_i(\boldsymbol{Z}_{t_i})}_{\text{reconstruction-based correction}}, \qquad i < M. \tag{15}$$

Thus, both updates consist of a base vector-field step followed by a term that steers the predicted endpoint. This shared algebraic form is the precise connection between the two methods.

The two methods, however, construct the steering term differently. FlowChef predicts the endpoint with the same vector field that drives the base step and steers it toward an externally specified objective L. Applying this construction directly to an image editing model means predicting the endpoint with the edit-prompt velocity $\boldsymbol{v}_i^{(c)}$ and steering it toward $\boldsymbol{X}_{\text{in}}$. This yields exactly the edit-prompt variant of Proposition 2 with $\alpha = 2s$, which rescales the response to the edit instruction. The proposed correction instead predicts the endpoint with the empty-prompt velocity $\boldsymbol{u}_i$ while the base step keeps $\boldsymbol{v}_i^{(c)}$. This separation of two conditions of the same editing model has no counterpart in FlowChef and is what preserves the coefficient h_i on $\boldsymbol{\delta}_i$. Moreover, $\tilde{L}_i$ is not a user-specified objective but is defined by the editing model's own input-conditioned prediction. Its unique minimizer

Table 1: Reconstruction under an empty prompt. SSIM and LPIPS are multiplied by 100.

Method	Full image			Text region	
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑
Baseline	11.84	48.02	41.13	8.71	28.90
Ours	**22.27**	**86.25**	**7.32**	**18.50**	**78.13**

Table 2: Quantitative comparison for text removal. SSIM and LPIPS are multiplied by 100.

Method	Model	Full image			Text region	
		PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑
Baseline	Qwen Image Edit 2509	19.62	72.76	7.16	19.07	60.77
Noise Inversion	Qwen Image Edit 2509	20.29	75.66	7.23	18.94	61.80
Ours	Qwen Image Edit 2509	22.07	80.61	6.00	20.04	64.15
FlowChef (Patel et al., 2025)	Qwen Image 2512	13.97	52.11	39.68	5.68	8.42
FlowEdit (Kulikov et al., 2025)	Qwen Image 2512	10.66	35.24	10.69	8.96	16.57
Baseline	FLUX.1 Kontext [dev]	26.42	90.47	6.60	21.00	66.97
Ours	FLUX.1 Kontext [dev]	**27.36**	93.81	**4.21**	**21.48**	**68.97**
FlowChef (Patel et al., 2025)	FLUX.1 [dev]	16.91	75.44	19.43	4.83	5.54
FlowEdit (Kulikov et al., 2025)	FLUX.1 [dev]	14.48	68.98	28.18	5.59	7.59
FireFlow (Deng et al., 2025)	FLUX.1 [dev]	10.62	40.48	57.70	7.35	12.99
RF-Edit (Wang et al., 2025)	FLUX.1 [dev]	12.05	48.71	50.54	7.79	14.84
RF-Inversion (Rout et al., 2025)	FLUX.1 [dev]	11.43	41.37	49.26	7.96	15.25
SickZil-Machine (Ko & Cho, 2020)	Task-specific CNNs	24.94	**95.30**	5.05	16.16	48.80

is the closed-form input-specific target $\boldsymbol{Z}^*_{t_i} = \boldsymbol{X}_{\text{in}} - \Delta_i \boldsymbol{u}_i$, toward which Algorithm 1 interpolates directly, and the gradient form above is the exact algebraic interpretation of this interpolation. Together with the early-step schedule, these choices define a distinct construction tailored to structure-preserving editing.

5 Experiments: Mask-Free Text Removal

Mask-free text removal aims to remove text written in an input manga image in a natural manner, without a segmentation mask for the text regions at inference time. This task is of high practical importance in manga localization, because text-free versions of the pages are often unavailable to translation teams. Older titles and back issues were typically produced without layered data that separate text from artwork. Even for recent titles, the layered files held by the authors are often not passed on to publishers and thus never reach the localization workflow. Moreover, hand-drawn text such as sound effects is drawn directly as part of the artwork. Text therefore has to be removed from the rasterized page itself. Automating this removal without manual masking substantially reduces the localization workload.

5.1 Dataset

Quantitative metrics such as PSNR, SSIM, and LPIPS require a text-free ground-truth image paired with each input. However, no publicly available manga text-removal dataset provides such pairs while satisfying the copyright and reproducibility criteria detailed in supplementary Sec. A. We therefore construct controlled synthetic pairs from Manga109s (Aizawa et al., 2020; Matsui et al., 2017).

Specifically, we constructed 3,204 pairs of input manga images with text and ground-truth manga images without text as follows. First, we used the text annotations in Manga109s to extract the largest rectangular crop without annotated text regions from each two-page manga spread. However, Manga109s also contains unannotated text such as handwritten text, so some cropped images could still include text. We therefore manually inspected the crops and removed those that still contained text. The resulting 3,204 cropped images were used as the ground-truth text-free manga images,

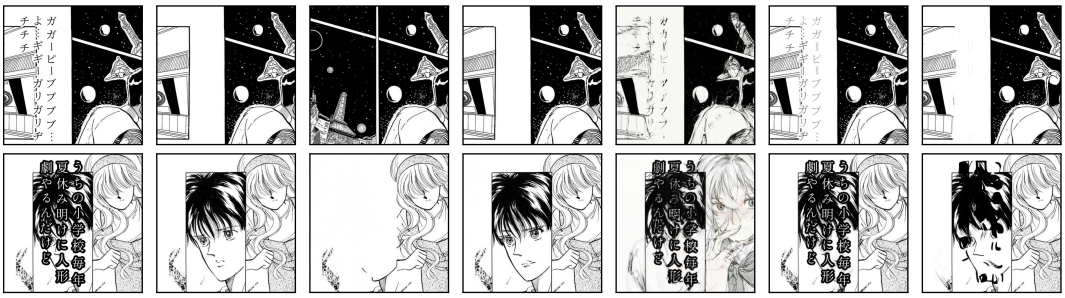


Figure 3: Qualitative comparison for text removal. Baseline and Ours use Qwen Image Edit 2509; FlowChef (Patel et al., 2025) and FlowEdit (Kulikov et al., 2025) use Qwen Image 2512; SickZil-Machine (Ko & Cho, 2020) uses its publicly released CNNs. ©Nagano Noriko ©Omi Ayuko

and the corresponding input images with text were created by rendering text on top of them. The rendered text content was randomly sampled from the text annotations originally present in the corresponding two-page spread before cropping. The font was randomly selected from 24 fonts that looked natural for manga images. Because the text is rendered synthetically, its exact pixel mask is known. All text-region PSNR and SSIM values use this ground-truth rendering mask rather than a mask estimated by a segmentation model, so the evaluation is independent of segmentation quality.

This construction provides exact paired ground truth and text-region masks, enabling a fair quantitative comparison in the absence of suitable real paired data.

5.2 Implementation Details

As the baseline, we used the open-source image editing model Qwen Image Edit 2509. All hyperparameters were kept at their default values. For our method, we set the number of corrected steps to $M = 3$ and the correction strength to $\alpha = 0.01$. We use this fixed setting for all main experiments and do not tune either hyperparameter per image using ground truth. For all methods and all images, we use the same fixed random seed (seed 0).

5.3 Preliminary Evaluation

As a sanity check, we first verify that the proposed method faithfully reconstructs the input image when the edit prompt is empty, i.e., $c = \emptyset$. Specifically, we evaluate the discrepancy between the image generated with $c = \emptyset$ and **the input image with rendered text**. As shown in Table 1, our method reconstructs the input image with text much more accurately than the baseline. The qualitative results in Fig. 1 further support this observation.

5.4 Comparison with Other Methods

We remove text by feeding an image with text and using the editing prompt "Remove text. Keep everything else unchanged.", and then measure the discrepancy between the generated edited image and **the ground-truth text-free image**. We do not include classification- or segmentation-oriented test-time adaptation methods as baselines. Their adaptation objectives are defined on class predictions or segmentation maps, which an image editing model does not produce, so they cannot be directly applied to this task.

We compare against the following methods:
Baseline: directly applying Qwen Image Edit 2509.
Noise Inversion: instead of generating the edited image from pure noise, this method starts generation from an intermediate point on the straight path connecting the noise $\boldsymbol{Z}_0$ and the input image $\boldsymbol{X}_{\text{in}}$. More specifically, generation starts from $(1 - t_i)\boldsymbol{Z}_0 + t_i\boldsymbol{X}_{\text{in}}$, which corresponds to advancing along the path for the first i steps. We report the best-performing setting, $i = 1$.
FlowChef (Patel et al., 2025) and FlowEdit (Kulikov et al., 2025): image editing methods based

on image generation models. Because they are not designed for instruction-based image editing models, they cannot be run on exactly the same backbone as ours. We therefore align the backbones as closely as possible. Their original implementations use FLUX.1 [dev], the image generation counterpart of FLUX.1 Kontext, and we additionally evaluate variants using Qwen Image 2512, from the same model family as Qwen Image Edit 2509. The source prompt is "A manga image with text." and the target prompt is "A manga image without text."

SickZil-Machine (Ko & Cho, 2020): a model designed and trained specifically for mask-free text removal in manga images. We use the publicly available pretrained weights.

FireFlow (Deng et al., 2025), RF-Edit (Wang et al., 2025), and RF-Inversion (Rout et al., 2025): recent inversion and trajectory-based editing methods that we evaluate with FLUX.1 [dev], the image generation counterpart of FLUX.1 Kontext, to provide a comparison with conceptually related trajectory methods.

Inpainting methods that require masks, such as Manga Inpainting (Xie et al., 2021), cannot be directly applied to this task.

Table 2 reports the quantitative comparison. For both Qwen Image Edit 2509 and FLUX.1 Kontext, our method improves all five metrics over the corresponding baseline, showing a consistent benefit over direct inference. With FLUX.1 Kontext, our method further obtains the best values among all compared methods on four of the five metrics, including both text-region metrics. With Qwen Image Edit 2509, our method also outperforms the task-specific SickZil-Machine within the exact ground-truth text regions, obtaining 20.04 versus 16.16 PSNR and 64.15 versus 48.80 SSIM. These gaps are substantial: because PSNR is logarithmic in MSE, the 3.88 dB difference corresponds to approximately $2.4\times$ lower MSE, and the SSIM difference is 15.35 points. The text-region scores use the ground-truth rendering masks and are therefore independent of SickZil-Machine's estimated segmentation. SickZil-Machine obtains higher full-image scores than the Qwen-based variant because it modifies only its segmented regions and copies all other pixels unchanged. Consequently, the unedited background dominates the full-image metrics even when text is missed or local completion is inaccurate. The supplementary material further verifies the effect of the empty prompt, shows that the method is not sensitive to its hyperparameters, and reports results with additional image editing models.

Figure 3 shows qualitative comparisons of the edited images. In the examples shown, the baseline edits regions more aggressively than necessary, whereas our method removes text while better preserving the overall structure of the input image. In these examples, SickZil-Machine leaves text behind when text-region segmentation fails and does not faithfully complete some facial regions.

5.5 Runtime and Memory

We evaluate both runtime and peak memory using FLUX.1 Kontext on an NVIDIA RTX A6000 with 48 GB of VRAM. We use FLUX.1 Kontext for this experiment because the baseline Qwen Image Edit 2509 pipeline, without the proposed correction, already exceeds the A6000's 48 GB memory capacity. With FLUX.1 Kontext, runtime increases from 52.39 to 58.23 seconds and peak memory from 33.83 to 33.85 GB, i.e., 11% runtime and 0.1% peak-memory overhead.

As an additional latency measurement, we evaluate Qwen Image Edit 2509 on an NVIDIA H200. The baseline takes an average of 35.56 seconds per image and our method takes 35.60 seconds over three images, with standard deviations of 0.01 and 0.03 seconds, respectively. This corresponds to a 0.1% runtime increase.

6 Conclusion

We presented a lightweight, training-free trajectory correction that anchors early editing steps to empty-prompt reconstruction, preserving global composition during local manga edits. Our frozen-velocity analysis shows how each correction step moves the estimated endpoint toward the input without weakening the editing response introduced by the prompt. Experiments with Qwen Image Edit 2509 and FLUX.1 Kontext indicate improved text removal over direct inference, with small practical overhead and no training, architectural changes, or additional sequential model passes. Future work will examine real text-removal data, other applications such as colorization, and adaptive anchoring for edits requiring larger structural or semantic changes.

AI USE STATEMENT

In this work, we used generative AI tools (LLM-based assistants) for three tasks with required disclosure. First, we used them as coding assistants when implementing the proposed method, the experiment scripts, and the preprocessing scripts that construct the evaluation pairs from Manga109s. The pairs themselves are produced by these scripts and are not generated by generative AI tools. Second, we used them to translate the draft written by the authors into English. Third, we used them to assist in deriving and checking the mathematical statements and proofs in Sec. 4, namely Propositions 1 and 2 and the relation between the proposed update and FlowChef in Sec. 4.3. We have not used generative AI tools for the other tasks with required disclosure, namely proposing the method, formulating hypotheses, designing experiments, and analyzing or interpreting results. Tasks related to surveys, interviews, or recordings are not applicable to this work. Additionally, for tasks with recommended disclosure, we used generative AI tools to edit the manuscript for readability and conciseness, including rephrasing and restructuring sentences and adjusting the LaTeX layout. We also used them to identify relevant recent literature and to check the bibliographic information of the references. We have reviewed all AI-assisted work. The authors reviewed and tested all AI-assisted code to confirm that it correctly implements the method, the dataset construction, and the experimental protocol described in the paper. The authors verified every derivation in Sec. 4 by hand. Before citing any paper found with the aid of AI tools, the authors checked it against the original paper and its official publication record. The authors also reviewed all translated and edited text to ensure that it accurately describes our method and results. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Because manga images are copyrighted artworks, all experiments use only Manga109s, whose manga authors have granted permission for its use in academic research (supplementary Sec. A). The proposed method is training-free: it adapts a pretrained image editing model to each input at inference time, so no manga images are collected or used to train or fine-tune any model. The manga images in the figures are taken from Manga109s and are shown with the copyright notices of their authors. We do not redistribute the derived evaluation pairs; instead, the submitted code includes the scripts that reconstruct them from Manga109s, which users obtain under its own terms. The synthetic text is rendered with openly licensed fonts. All pretrained models were used in accordance with their respective licenses. This work does not involve human subjects or crowdsourced annotation; the manual inspection of the crops was performed by the authors.

Text removal is a key step in manga localization, and the same capability could also be used to produce unauthorized translations or to remove credits and copyright notices. The proposed correction does not add a new editing capability to the base model; it constrains the editing trajectory of an existing, publicly available model so that the input's structure is preserved. The intended use is localization by rights holders and authorized translation teams, as in the practical scenario described in Sec. 5, and any deployment should respect the rights of the original creators. Moreover, because the method involves no training, it does not extract the style of any specific artist from manga data.

REPRODUCIBILITY STATEMENT

The proposed correction is specified by Algorithm 1 and Sec. 3.3. The fixed hyperparameters and random seed used in all main experiments are given in Sec. 5.2, and their sensitivity is analyzed in supplementary Sec. B.3. The editing prompts, backbones, and settings of the compared methods are given in Sec. 5.4 and supplementary Sec. C.2, and the additional backbones in supplementary Sec. B.4. The construction of the text-removal and screentone-synthesis datasets from Manga109s is described step by step in Sec. 5.1 and supplementary Sec. C.1. The hardware used for the runtime and memory measurements is specified in Sec. 5.5. The propositions in Sec. 4 state their frozen-velocity assumption explicitly and are accompanied by complete proofs. As supplementary material, we submit anonymized source code that implements the proposed sampler for Qwen Image Edit and FLUX.1 Kontext and provides demos for both tasks. The code also includes the preprocessing scripts that reconstruct both evaluation datasets from Manga109s, which researchers can obtain under its own terms of use. Because the text-removal crops were screened manually, these scripts include the annotation tool used for the screening.

REFERENCES

Kiyoharu Aizawa, Azuma Fujimoto, Atsushi Otsubo, Toru Ogawa, Yusuke Matsui, Koki Tsubota, and Hikaru Ikuta. Building a manga dataset "manga109" with annotations for multimedia applications. *IEEE MultiMedia*, 27(2):8–18, 2020. doi: 10.1109/mmul.2020.2987895.

Omri Avrahami, Or Patashnik, Ohad Fried, Egor Nemchinov, Kfir Aberman, Dani Lischinski, and Daniel Cohen-Or. Stable Flow: Vital layers for training-free image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7877–7888, 2025.

Kecheng Chen, Xinyu Luo, Tiexin Qin, Jie Liu, Hui Liu, Victor Ho Fun Lee, Hong Yan, and Haoliang Li. Test-time adaptation for foundation medical segmentation model without parametric updates. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20075–20084, 2025.

Yingying Deng, Xiangyu He, Changwang Mei, Peisong Wang, and Fan Tang. Fireflow: Fast inversion of rectified flow for image semantic editing. In *International Conference on Machine Learning*, pp. 13110–13128. PMLR, 2025.

Chie Furusawa, Kazuyuki Hiroshiba, Keisuke Ogaki, and Yuri Odagiri. Comicolorization: semi-automatic manga colorization. In *SIGGRAPH Asia 2017 Technical Briefs*, pp. 1–4. 2017.

Maksim Golyadkin, Ianis Plevokas, and Ilya Makarov. Closing the domain gap in manga colorization via aligned paired dataset. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 5580–5590. IEEE, 2025.

Jiayi Guo, Junhao Zhao, Chaoqun Du, Yulin Wang, Chunjiang Ge, Zanlin Ni, Shiji Song, Humphrey Shi, and Gao Huang. Everything to the synthetic: Diffusion-driven test-time adaptation via synthetic-domain alignment. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 30503–30513, 2025.

Paulina Hensman and Kiyoharu Aizawa. cgan-based manga colorization using a single training image. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 3, pp. 72–77. IEEE, 2017.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Vincent Tao Hu, Wei Zhang, Meng Tang, Pascal Mettes, Deli Zhao, and Cees Snoek. Latent space editing in transformer-based flow matching. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 2247–2255, 2024.

Yi Huang, Jiancheng Huang, Yifan Liu, Mingfu Yan, Jiaxi Lv, Jianzhuang Liu, Wei Xiong, He Zhang, Liangliang Cao, and Shifeng Chen. Diffusion model-based image editing: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Yuusuke Kataoka, Takashi Matsubara, and Kuniaki Uehara. Automatic manga colorization with color style by generative adversarial nets. In *2017 18th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, pp. 495–499. IEEE, 2017.

Jimyeong Kim, Jungwon Park, Yeji Song, Nojun Kwak, and Wonjong Rhee. ReFlex: Text-guided editing of real images in rectified flow via mid-step feature extraction and attention adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15939–15948, 2025.

U-Ram Ko and Hwan-Gue Cho. Sickzil-machine: a deep learning based script text isolation system for comics translation. In *International Workshop on Document Analysis Systems*, pp. 413–425. Springer, 2020.

Vladimir Kulikov, Matan Kleiner, Inbar Huberman-Spiegelglas, and Tomer Michaeli. FlowEdit: Inversion-free text-based editing using pre-trained flow models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19721–19730, 2025.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. FLUX. 1 Kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025.

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

Jian Lin, Xueting Liu, Chengze Li, Minshan Xie, and Tien-Tsin Wong. Sketch2manga: Shaded manga screening from sketch with diffusion models. In *2024 IEEE International Conference on Image Processing (ICIP)*, pp. 2389–2395. IEEE, 2024.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Shiyu Liu, Yucheng Han, Peng Xing, Fukun Yin, Rui Wang, Wei Cheng, Jiaqi Liao, Yingming Wang, Honghao Fu, Chunrui Han, et al. Step1x-edit: A practical framework for general image editing. *arXiv preprint arXiv:2504.17761*, 2025.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023.

Xiaocheng Lu, Jingcai Guo, and Song Guo. Consistent-Inversion: Reverse consistency guidance for structure-preserving visual editing. *arXiv preprint arXiv:2606.07145*, 2026.

Yuhang Ma, Bo Cheng, Shanyuan Liu, Hongyi Zhou, Liebucha Wu, Xiaoyu Wu, Dawei Leng, and Yuhui Yin. Nami: Efficient image generation via bridged progressive rectified flow transformers. *arXiv preprint arXiv:2503.09242*, 2025.

Yusuke Matsui, Toshihiko Yamasaki, and Kiyoharu Aizawa. Interactive manga retargeting. In *ACM SIGGRAPH 2011 Posters*, pp. 1–1. 2011.

Yusuke Matsui, Kota Ito, Yuji Aramaki, Azuma Fujimoto, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa. Sketch-based manga retrieval using manga109 dataset. *Multimedia Tools and Applications*, 76(20):21811–21838, 2017. doi: 10.1007/s11042-016-4020-z.

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

Maitreya Patel, Song Wen, Dimitris N Metaxas, and Yezhou Yang. FlowChef: Steering of rectified flow models for controlled generations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15308–15318, 2025.

Mihir Prabhudesai, Tsung-Wei Ke, Alex Li, Deepak Pathak, and Katerina Fragkiadaki. Diffusion-tta: Test-time adaptation of discriminative models via generative feedback. *Advances in Neural Information Processing Systems*, 36:17567–17583, 2023.

Yurui Qian, Qi Cai, Yingwei Pan, Yehao Li, Ting Yao, Qibin Sun, and Tao Mei. Boosting diffusion models with moving average sampling in frequency domain. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8911–8920, 2024.

Xiangyan Qu, Zhenlong Yuan, Jing Tang, Rui Chen, Datao Tang, Meng Yu, Lei Sun, Yancheng Bai, Xiangxiang Chu, Gaopeng Gou, Gang Xiong, and Yujun Cai. From scale to speed: Adaptive test-time scaling for image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Yingge Qu, Tien-Tsin Wong, and Pheng-Ann Heng. Manga colorization. *ACM Transactions on Graphics (ToG)*, 25(3):1214–1220, 2006.

Litu Rout, Yujia Chen, Nataniel Ruiz, Constantine Caramanis, Sanjay Shakkottai, and Wen-Sheng Chu. Semantic image inversion and editing using rectified stochastic differential equations. In *International Conference on Learning Representations*, 2025.

Kazuhiro Sato, Yusuke Matsui, Toshihiko Yamasaki, and Kiyoharu Aizawa. Reference-based manga colorization by graph correspondence using quadratic programming. In *SIGGRAPH Asia 2014 Technical Briefs*, pp. 1–4. 2014.

Yugo Shimizu, Ryosuke Furuta, Delong Ouyang, Yukinobu Taniguchi, Ryota Hinami, and Shonosuke Ishiwatari. Painting style-aware manga colorization based on generative adversarial networks. In *2021 IEEE International Conference on Image Processing (ICIP)*, pp. 1739–1743. IEEE, 2021.

Meituan LongCat Team, Hanghang Ma, Haoxian Tan, Jiale Huang, Junqiang Wu, Jun-Yan He, Lishuai Gao, Songlin Xiao, Xiaoming Wei, Xiaoqi Ma, et al. Longcat-image technical report. *arXiv preprint arXiv:2512.07584*, 2025.

Yun-Yun Tsai, Fu-Chen Chen, Albert YC Chen, Junfeng Yang, Che-Chun Su, Min Sun, and Cheng-Hao Kuo. Gda: Generalized diffusion for robust test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23242–23251, 2024.

Koki Tsubota, Daiki Ikami, and Kiyoharu Aizawa. Synthesis of screentone patterns of manga characters. In *2019 IEEE international symposium on multimedia (ISM)*, pp. 212–2123. IEEE, 2019.

Binxu Wang and John J Vastola. Diffusion models generate images like painters: an analytical theory of outline first, details later. In *arXiv preprint arXiv:2303.02490*, 2023.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021.

Jiangshan Wang, Junfu Pu, Zhongang Qi, Jiayi Guo, Yue Ma, Nisha Huang, Yuxin Chen, Xiu Li, and Ying Shan. Taming rectified flow for inversion and editing. In *International Conference on Machine Learning*, pp. 64044–64058. PMLR, 2025.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025.

Minshan Xie, Menghan Xia, Xueting Liu, Chengze Li, and Tien-Tsin Wong. Seamless manga inpainting with semantics awareness. *ACM Transactions on Graphics (TOG)*, 40(4):1–11, 2021.

Minshan Xie, Menghan Xia, Chengze Li, Xueting Liu, and Tien-Tsin Wong. Screentone-preserved manga retargeting. In *Computer Graphics Forum*, volume 44, pp. e70096. Wiley Online Library, 2025.

Yue Yu, Yang Jiao, Jiayu Wang, Qi Dai, and Jingjing Chen. Adaptive inference-time scaling via early-step latent verification for image editing. *arXiv preprint arXiv:2606.15188*, 2026.

INFERENCE-TIME TRAJECTORY CORRECTION FOR STRUCTURE-PRESERVING MANGA IMAGE EDITING

SUPPLEMENTARY MATERIAL

A EVALUATION SCOPE

Our evaluation is designed around practical manga-editing tasks and data that satisfy both copyright-provenance and reproducibility criteria. Because manga images are copyrighted artworks, we do not regard online availability alone as authorization to use commercial manga for training or evaluation. Under our dataset-selection policy, we require documented permission for research use. Manga109s (Aizawa et al., 2020; Matsui et al., 2017) meets this criterion because its manga authors formally granted such permission, and its distribution supports reproducible evaluation.

We evaluate text removal and screentone synthesis using Manga109s. These practically important tasks are complementary: text removal deletes existing content, whereas screentone synthesis adds new content to line art. Both require the global manga structure to be preserved, and controlled preprocessing of Manga109s allows pairs of input and ground-truth images needed for quantitative evaluation to be constructed.

These criteria also guide our treatment of other datasets. For existing large-scale paired colorization data (Golyadkin et al., 2025), the available documentation did not establish copyright provenance to the level required by our policy. MangaZero (Wu et al., 2025) presents a separate reproducibility issue: its public release relies on source-image URLs, some of which are unavailable, making complete reconstruction of the released dataset difficult. We therefore do not use either source for training or evaluation. Semantic edits such as changing character attributes pose a different challenge because objective paired ground truth is difficult to construct. The selected tasks thus provide reproducible quantitative benchmarks for both removing and adding content while preserving global manga structure.

B ADDITIONAL EXPERIMENTS: MASK-FREE TEXT REMOVAL

Unless otherwise noted, all evaluations in this section measure the discrepancy between the generated edited image and the ground-truth text-free image.

B.1 EFFECT OF THE PROMPT USED FOR TARGET CONSTRUCTION

To verify the effect of using an empty prompt, we evaluate a variant of Eq. (5) in which the edit prompt c is used instead of the empty prompt $\emptyset$. For all results in this comparison, we measure the discrepancy between the generated edited image and the ground-truth text-free image. Table 3 shows that both variants improve the full-image metrics over the baseline, confirming that anchoring the early trajectory toward the input preserves the global composition. However, only the empty-prompt target improves both text-region metrics: with the edit prompt, text-region PSNR falls below the baseline (18.69 vs. 19.07 dB), whereas the empty prompt improves it to 20.04 dB. As predicted by Proposition 2, the edit-prompt target reduces the coefficient of the prompt-induced term from h_i to $h_i - \alpha\Delta_i$, pulling the output back toward the input and leaving text behind (Fig. 4). The empty prompt thus retains the global-preservation effect of the anchor without suppressing the requested edit.

B.2 EFFECT OF THE RECONSTRUCTION TARGET

We next analyze the effect of the reconstruction target computed by Eqs. (4) and (5). Instead of Eqs. (4) and (5), we also consider correcting the trajectory using the straight path between the noise $\boldsymbol{Z}_0$ and the input image $\boldsymbol{X}_{\text{in}}$:

$$\boldsymbol{Z}^*_{t_i} = (1 - t_i)\boldsymbol{Z}_0 + t_i\boldsymbol{X}_{\text{in}}. \tag{16}$$

Table 3: Effect of the prompt used for target construction. SSIM and LPIPS are multiplied by 100.

Method	Full image			Text region	
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑
Baseline	19.62	72.76	7.16	19.07	60.77
Edit prompt	21.57	**80.68**	6.74	18.69	61.99
Empty prompt	**22.07**	80.61	**6.00**	**20.04**	**64.15**



Input image Edit prompt Empty prompt

Figure 4: Effect of the prompt used for target construction. ©Okuda Momoko

Trajectory correction is then performed in the same way as Eq. (6).

Table 4 summarizes the results. Using the reconstruction target yields better point estimates on all five metrics than correcting the trajectory with the straight path. Because Rectified Flow models are trained using straight paths as supervision, the reconstruction target should coincide with the straight path if a pretrained Rectified Flow model behaves ideally. However, on data that are rarely seen during training, such as manga images, Rectified Flow models may deviate from this ideal behavior, making the reconstruction target more effective in the evaluated setting.

B.3 Effect of Hyperparameters

Table 5 serves two purposes: it verifies stability within a practical operating range and probes deliberately extreme settings to clarify the role of each hyperparameter. Within the practical ranges examined, the proposed method improves all five metrics over the baseline for all tested values of M between 1 and 7 and of α between 0.01 and 0.05. The method therefore does not require sensitive tuning within these ranges; stable improvements are obtained by correcting a small number of early steps and keeping α within the same order of magnitude as the default.

The settings below the separator are deliberately outside the practical ranges and are included to interpret the role of each hyperparameter, rather than as recommended operating points. Setting $M = 9$ extends the correction into later timesteps and makes the output overly close to the input: full-image PSNR and SSIM improve, but residual text degrades the text-region metrics. This result illustrates why the method is designed to correct only the early trajectory. Similarly, $\alpha = 0.001$ and $\alpha = 0.1$ are respectively one tenth and ten times the default value. The former is too weak to sufficiently correct the trajectory, whereas the latter over-corrects toward the input and degrades LPIPS and the text-region metrics. These deliberately extreme values illustrate the expected under- and over-correction behavior; they do not indicate fragility within the practical range.

For all main experiments, we use the same fixed setting, $M = 3$ and $\alpha = 0.01$, without per-image tuning using ground truth.

Table 4: Effect of the reconstruction target. SSIM and LPIPS are multiplied by 100.

Method	Full image			Text region	
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑
Baseline	19.62	72.76	7.16	19.07	60.77
Straight path	19.69	73.13	7.74	19.13	61.12
Reconstruction target	**22.07**	**80.61**	**6.00**	**20.04**	**64.15**

Table 5: Stability within practical ranges and interpretation using deliberately extreme hyperparameter settings. SSIM and LPIPS are multiplied by 100.

Method	Hyper-param.	Full image			Text region	
		PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑
Baseline		19.62	72.76	7.16	19.07	60.77
Practical operating range						
Ours	$M = 1$	20.89	76.87	6.94	19.71	62.56
	$M = 3$	22.07	80.61	6.00	20.04	64.15
	$M = 5$	22.31	81.67	6.01	19.82	63.59
	$M = 7$	22.37	82.42	6.45	19.23	61.31
	$\alpha = 0.01$	22.07	80.61	6.00	20.04	64.15
	$\alpha = 0.02$	23.40	84.38	5.30	20.54	64.97
	$\alpha = 0.03$	24.39	87.43	5.10	20.76	64.49
	$\alpha = 0.04$	24.73	89.03	5.30	20.36	62.60
	$\alpha = 0.05$	24.87	89.84	5.46	20.25	61.65
Deliberately extreme settings						
Ours	$M = 9$	22.38	83.10	6.93	18.48	57.92
	$\alpha = 0.001$	19.67	72.92	7.87	19.08	60.83
	$\alpha = 0.1$	22.74	86.64	9.76	18.40	54.58

B.4 Applicability to Various Image Editing Models

Table 6 compares the correction across all four image editing backbones evaluated for text removal. The correction improves all five metrics for Qwen Image Edit 2509 and FLUX.1 Kontext. For Qwen Image Edit 2511 and FLUX.2 Klein 9B, four of the five metrics improve only slightly, while text-region SSIM and PSNR, respectively, decrease slightly. These results show that the correction can be applied to multiple backbones without model-specific training, although the magnitude of the quantitative changes varies across them. The improvement is largest for Qwen Image Edit 2509, whose baseline shows the largest structural deviation from the input (full-image SSIM 72.76). For FLUX.2 Klein 9B, whose baseline already attains a full-image SSIM of 94.67, the room for improvement is inherently small. Even on backbones with small gains, the correction does not noticeably degrade any metric (the largest decrease is 0.31 points in text-region SSIM), so it can be applied without backbone-specific tuning.

C Experiments: Screentone Synthesis

C.1 Dataset

Screentone synthesis takes a line-art image as input and outputs a final manga image with screentones. Following Tsubota et al. (2019), we constructed 1,355 pairs of input line art and ground-truth manga images with screentones as follows. We cropped character bounding boxes from six titles in

Table 6: Evaluation across image editing models for text removal. SSIM and LPIPS are multiplied by 100.

Method	Model	Full image			Text region	
		PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑
Baseline	Qwen Image Edit 2509	19.62	72.76	7.16	19.07	60.77
Ours	Qwen Image Edit 2509	**22.07**	**80.61**	**6.00**	**20.04**	**64.15**
Baseline	Qwen Image Edit 2511	21.19	82.63	10.16	18.51	**62.86**
Ours	Qwen Image Edit 2511	**21.39**	**83.05**	**9.84**	**18.54**	62.55
Baseline	FLUX.1 Kontext [dev]	26.42	90.47	6.60	21.00	66.97
Ours	FLUX.1 Kontext [dev]	**27.36**	**93.81**	**4.21**	**21.48**	**68.97**
Baseline	FLUX.2 Klein 9B	26.90	94.67	3.48	**19.72**	66.32
Ours	FLUX.2 Klein 9B	**27.07**	**94.91**	**3.44**	19.61	**66.38**

Manga109s to obtain 1,355 ground-truth manga images with screentones. We then converted them into input line-art images using Manga Line Extraction (Li et al., 2017).

C.2 Comparison Methods

As the baseline model, we use Qwen Image Edit 2511. The editing prompt is "Convert the line art into a finished Japanese black and white monochrome manga image." We also apply the proposed correction to FLUX.1 Kontext and FLUX.2 Klein 2B. For all three models, we use the same hyperparameters as in the text-removal experiments ($M = 3$ and $\alpha = 0.01$).

We additionally compare with Noise Inversion, FlowChef (Patel et al., 2025), and FlowEdit (Kulikov et al., 2025). As in the text-removal experiments, FlowChef and FlowEdit are evaluated with Qwen Image 2512 and FLUX.1 [dev]. For FlowChef and FlowEdit, the source prompt is "A line art image." and the target prompt is "A finished Japanese black and white monochrome manga image."

The task-specific methods most directly relevant to this setting are Sketch2Manga (Lin et al., 2024) and the earlier method of Tsubota et al. (2019). Sketch2Manga is the more recent method and provides public pretrained weights, enabling reproducible evaluation. The method of Tsubota et al. requires screentone pattern data at inference time, but these data remained unavailable even after we contacted the authors. We therefore use Sketch2Manga as the task-specific baseline; as the more recent and reproducible of the two methods, it provides an appropriate comparison with a model designed specifically for screentone synthesis.

C.3 Results

The proposed correction is not designed to improve the screentone-synthesis ability of the underlying image editing model itself. Its role is to suppress unintended structural changes, thereby preserving the input's global composition, including its line-art structure, while the image editing model performs screentone synthesis.

Quantitatively evaluating this effect is challenging. Screentone synthesis has no unique correct output because many screentone placements can produce a plausible finished manga image, and the ground-truth image constructed in Sec. C.1 is only one such realization. Paired pixel-wise metrics such as PSNR and SSIM are therefore unsuitable. Since no standard metric evaluates both input-structure preservation and screentone quality, we use FID and CMMD (Jayasumana et al., 2024), which are commonly used to evaluate image generation, to check whether the outputs remain realistic at the distribution level. However, these metrics do not measure whether each output preserves the structure of its input, so the intended effect of the correction may not be fully reflected in them. We therefore assess structure preservation through qualitative comparisons.

Table 7 shows that the correction does not reduce realism for any of the three image editing models. FID and CMMD remain nearly unchanged for Qwen Image Edit 2511 (36.08 to 36.07 and 0.3140 to 0.3129, respectively) and improve for FLUX.1 Kontext and FLUX.2 Klein 2B, most notably by 9.28 points in FID for FLUX.1 Kontext.

Table 7: Quantitative evaluation of screentone synthesis.

Method	Model	FID ↓	CMMD ↓
Baseline	Qwen Image Edit 2511	36.08	0.3140
Noise Inversion	Qwen Image Edit 2511	36.71	**0.3123**
Ours	Qwen Image Edit 2511	**36.07**	0.3129
FlowChef	Qwen Image 2512	81.88	0.9993
FlowEdit	Qwen Image 2512	61.32	0.4669
Baseline	FLUX.1 Kontext [dev]	92.97	0.3746
Ours	FLUX.1 Kontext [dev]	83.69	0.3704
FlowChef	FLUX.1 [dev]	93.46	0.3928
FlowEdit	FLUX.1 [dev]	68.28	1.3403
Baseline	FLUX.2 Klein 2B	47.22	0.5630
Ours	FLUX.2 Klein 2B	46.89	0.5547
Sketch2Manga	Fine-tuned SD1.5	139.09	2.1467

Figure 5 provides qualitative comparisons. FlowChef, FlowEdit, and Sketch2Manga do not produce proper screentones on these inputs. The baseline adds screentones but frequently redraws the input: it alters faces and expressions, adds backgrounds or objects absent from the input, and in some cases rewrites the text in speech balloons. In contrast, our method adds screentones while retaining the input's characters, panel layout, and text. As noted above, these structural deviations are largely invisible to FID and CMMD. Because these metrics compare the set of outputs with the set of ground-truth images, a redrawn output that still looks like finished manga is not penalized. Figure 5 therefore complements Table 7 by showing the structure preservation that these metrics do not capture.



Figure 5: Qualitative comparison for screentone synthesis. Baseline and Ours use Qwen Image Edit 2511; FlowChef (Patel et al., 2025) and FlowEdit (Kulikov et al., 2025) use Qwen Image 2512; Sketch2Manga (Lin et al., 2024) uses its publicly released model fine-tuned from Stable Diffusion 1.5. ©Saki Kaori ©Kurita Riku ©Yabuno Tenya, Watanabe Tatsuya ©Uchida Minako

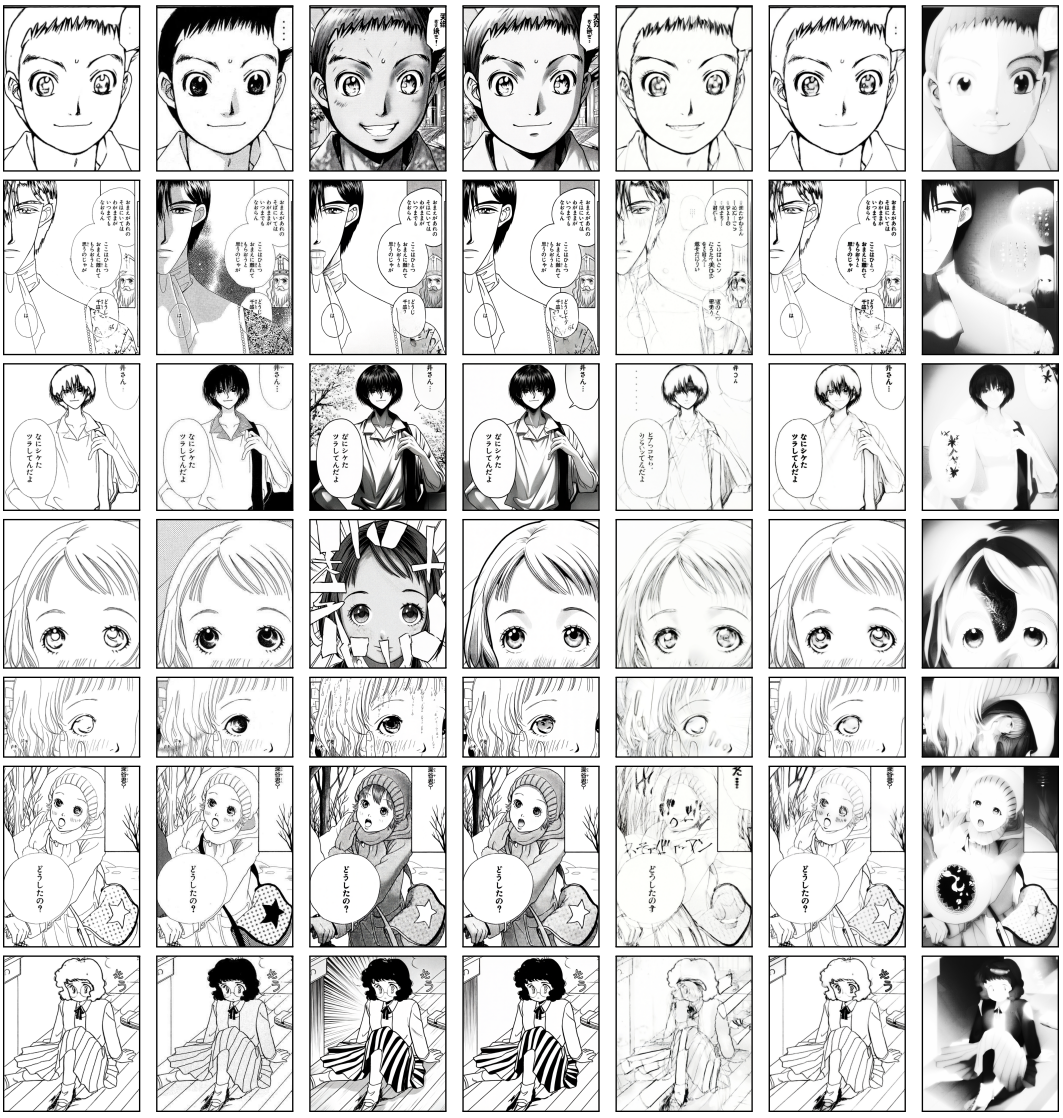


Figure 5: Qualitative comparison for screentone synthesis (continued). ©Saki Kaori ©Hanada Sakumi ©Yamada Uduki ©Kurita Riku

REFERENCES

Kiyoharu Aizawa, Azuma Fujimoto, Atsushi Otsubo, Toru Ogawa, Yusuke Matsui, Koki Tsubota, and Hikaru Ikuta. Building a manga dataset "manga109" with annotations for multimedia applications. *IEEE MultiMedia*, 27(2):8–18, 2020. doi: 10.1109/mmul.2020.2987895.

Maksim Golyadkin, Ianis Plevokas, and Ilya Makarov. Closing the domain gap in manga colorization via aligned paired dataset. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 5580–5590. IEEE, 2025.

Sadeep Jayasumana, Srikumar Ramalingam, Andreas Veit, Daniel Glasner, Ayan Chakrabarti, and Sanjiv Kumar. Rethinking fid: Towards a better evaluation metric for image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9307–9315, 2024.

Vladimir Kulikov, Matan Kleiner, Inbar Huberman-Spiegelglas, and Tomer Michaeli. FlowEdit: Inversion-free text-based editing using pre-trained flow models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19721–19730, 2025.

Chengze Li, Xueting Liu, and Tien-Tsin Wong. Deep extraction of manga structural lines. *ACM Transactions on Graphics (TOG)*, 36(4):1–12, 2017.

Jian Lin, Xueting Liu, Chengze Li, Minshan Xie, and Tien-Tsin Wong. Sketch2manga: Shaded manga screening from sketch with diffusion models. In *2024 IEEE International Conference on Image Processing (ICIP)*, pp. 2389–2395. IEEE, 2024.

Yusuke Matsui, Kota Ito, Yuji Aramaki, Azuma Fujimoto, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa. Sketch-based manga retrieval using manga109 dataset. *Multimedia Tools and Applications*, 76(20):21811–21838, 2017. doi: 10.1007/s11042-016-4020-z.

Maitreya Patel, Song Wen, Dimitris N Metaxas, and Yezhou Yang. FlowChef: Steering of rectified flow models for controlled generations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15308–15318, 2025.

Koki Tsubota, Daiki Ikami, and Kiyoharu Aizawa. Synthesis of screentone patterns of manga characters. In *2019 IEEE international symposium on multimedia (ISM)*, pp. 212–2123. IEEE, 2019.

Jianzong Wu, Chao Tang, Jingbo Wang, Yanhong Zeng, Xiangtai Li, and Yunhai Tong. Diffsensei: Bridging multi-modal llms and diffusion models for customized manga generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 28684–28693, 2025.