
Appendix to "DNAEdit: Direct Noise Alignment for Text-Guided Rectified Flow Editing"

In this appendix, we provide the following materials:

- A More details of the DNA algorithms (referring to Sec. 3.2 in the main paper);
- B Theoretical analysis for rectified flow (RF)-based editing methods;
- C Detailed experimental setup, including more implementation details and the construction details of DNA-Bench (referring to Sec. 4.1 in the main paper);
- D More editing results of DNAEdit and more visual comparisons on both PIE-bench and DNA-bench (referring to Sec. 4.2 in the main paper);
- E Ablation studies on DNAEdit (referring to Sec. 4.3 in the main paper);
- F The results of applying DNAEdit to video editing;
- G Broader impacts.

A More Details of the DNA Algorithm

A.1 Detailed Derivation of the DNA Algorithm

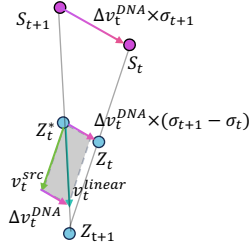


Figure 1: Direct noise alignment (DNA) for timestep t .

Assuming the noise is at S_{t+1} , the linear velocity v^{linear} at this point is given by $\frac{S_{t+1} - Z_{t+1}}{\sigma_{t+1}}$, while the velocity conditioned on source prompt v_t^{src} is represented by $v_\theta(Z_t^*, \psi^{src})$. To align the linear velocity with v_t^{src} , we move Z_t^* and S_{t+1} . Then Z_t can be derived by Eq. (1) using the velocity difference $\Delta v_t^{DNA} = \frac{Z_{t+1} - Z_t^*}{\sigma_{t+1} - \sigma_t} - v_t^{src}$:

$$Z_t = Z_t^* + \Delta v_t^{DNA}(\sigma_{t+1} - \sigma_t). \quad (1)$$

Accordingly, we need to shift the current Gaussian noise S_{t+1} to obtain a new Gaussian noise S_t to ensure that S_t , Z_t and Z_{t+1} still satisfy the properties of linear interpolation. Therefore, the ratio between S_t , Z_t and Z_{t+1} should adhere to the ratio specified in Eq. (2):

$$\begin{aligned} \frac{S_t - S_{t+1}}{Z_t - Z_t^*} &= \frac{\sigma_{t+1}}{\sigma_{t+1} - \sigma_t}, \\ S_t - S_{t+1} &= \frac{\sigma_{t+1}}{\sigma_{t+1} - \sigma_t} \times (Z_t - Z_t^*), \\ S_t - S_{t+1} &= \Delta v_t^{DNA} \times \sigma_{t+1}. \end{aligned} \quad (2)$$

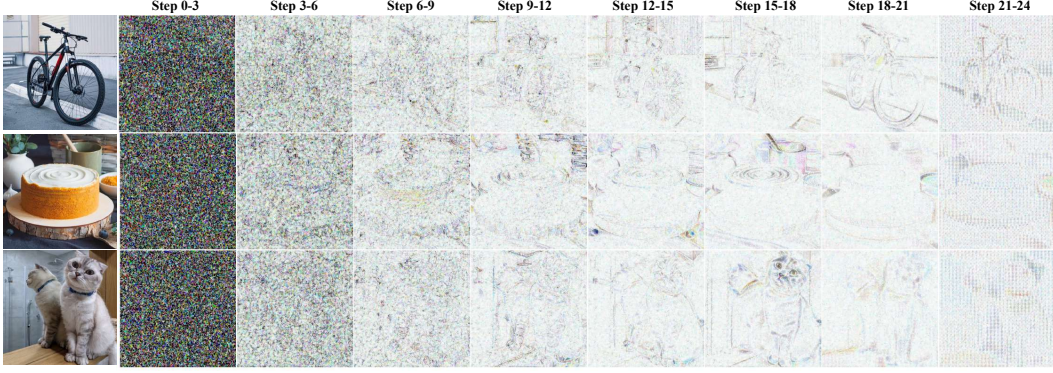
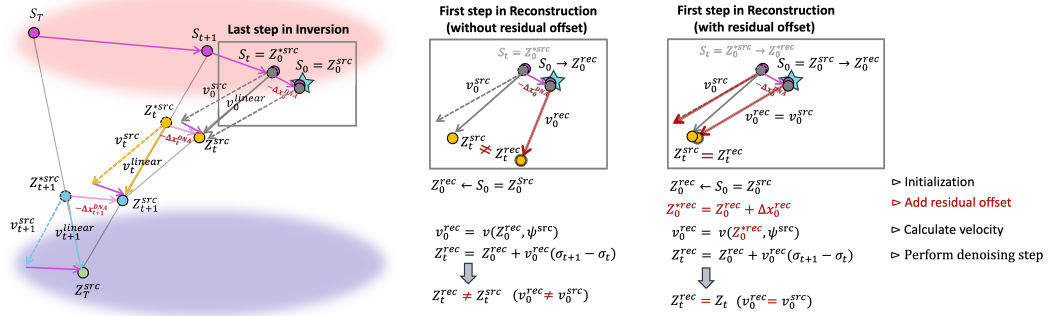


Figure 2: More visualization of the difference in Gaussian noise between S_t and S_{t+3} .

Through straightforward derivation, we can obtain that the shift required to move S_{t+1} to S_t is $\Delta v_t^{DNA} \times \sigma_{t+1}$ at timestep t .



(a) DNAEdit inversion (left) and the first step of reconstruction (right)

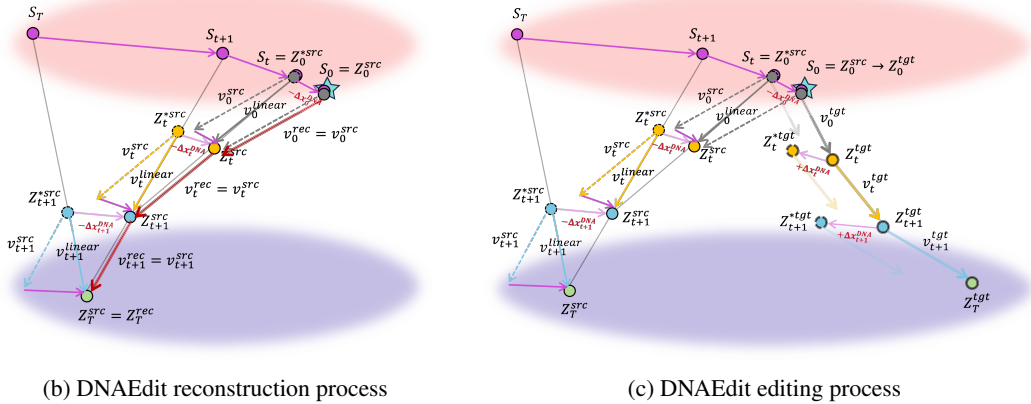


Figure 3: DNA reconstruction and editing without and with residual offset.

We provide additional visualizations in Fig. 2 to demonstrate the changes of noise S_t within the Gaussian noise space. Specifically, the noise difference $S_t - S_{t+3}$ is presented. It can be seen that as the DNA process progresses, the reference image's content is injected into the Gaussian noise through the difference in the velocity field.

Algorithm 1 Reconstruction Process with Residual Offset

Input: Source image Z_T^{src} , Timesteps $\{\sigma_t\}_{t=0}^T$, RF v_θ , Source text ψ^{src} and target text ψ^{tgt}
Output: Reconstructed image Z_T^{tgt}
Inversion Init: Random Gaussian noise S_T
for $t = T - 1, \dots, 0$ **do**
 $Z_t^{*src} \leftarrow Z_{t+1}^{src} \times \frac{\sigma_t}{\sigma_{t+1}} + S_{t+1} \times (1 - \frac{\sigma_t}{\sigma_{t+1}})$
 $v_t^{src} = v_\theta(Z_t^{*src}, \psi^{src})$
 $v_t^{linear} = (Z_{t+1}^{src} - S_{t+1}) / \sigma_{t+1}$
 $\Delta v_t^{DNA} = v_t^{linear} - v_t^{src}$
 $S_t \leftarrow S_{t+1} + \Delta v_t^{DNA} \times \sigma_{t+1}$
 $Z_t^{src} \leftarrow Z_t^{*src} + \Delta v_t^{DNA} \times (\sigma_{t+1} - \sigma_t)$
 $\Delta x_t^{DNA} = Z_t^{*src} - Z_t^{src}$
end for
Reconstruction Init: $Z_0^{rec} \leftarrow S_0 = Z_0^{src}$
for $t = 0, \dots, T - 1$ **do**
 $Z_t^{*rec} = Z_t^{rec} + \Delta x_t^{DNA} \Rightarrow Z_t^{*src}$ ▷ Add the residual offset back
 $v_t^{rec} = v_\theta(Z_t^{*rec}, \psi^{src}) \Rightarrow v_\theta(Z_t^{*src}, \psi^{rec}) = v_t^{src}$ ▷ Calculate velocity v_t^{rec} using source text
 $Z_{t+1}^{rec} = Z_t^{rec} - v_t^{rec} \times (\sigma_{t+1} - \sigma_t) \Rightarrow Z_{t+1}^{src}$
end for
return $Z_T^{rec} \Rightarrow Z_T^{src}$

A.2 DNA Reconstruction without and with Residual Offset

Fig. 3a illustrates the inversion and reconstruction process, comparing the effect of adding the residual offset. The left side shows the standard inversion path (timesteps: $T \rightarrow t + 1 \rightarrow t \rightarrow 0$), while the right side highlights the difference between reconstructions without and with the residual offset (for clarity, only the first step $t = 0$ is visualized). Without the residual offset, the reconstructed latent Z_t^{rec} deviates due to the mismatched velocity directions between v_0^{rec} and v_0^{src} , leading to accumulated error. In contrast, reintroducing the residual offset aligns each timestep’s velocity computation with the same noise latent used during inversion, which can be formulated as:

$$Z_t^{*rec} = Z_t^{rec} + \Delta x_t^{DNA} \Rightarrow Z_t^{*src}. \quad (3)$$

This corrected latent Z_t^{*rec} ensures that the velocity field can be computed consistently:

$$v_t^{rec} = v_\theta(Z_t^{*rec}, \psi^{src}) = v_\theta(Z_t^{*src}, \psi^{rec}) \Rightarrow v_t^{src} \quad (4)$$

As a result, the reconstruction can be formulated as:

$$Z_{t+1}^{rec} = Z_t^{rec} - v_t^{rec} \times (\sigma_{t+1} - \sigma_t) \Rightarrow Z_t^{rec} - v_t^{src} \times (\sigma_{t+1} - \sigma_t) = Z_{t+1}^{src}. \quad (5)$$

It becomes perfectly aligned with the original trajectory, eliminating cumulative errors and restoring the source latent precisely. This mechanism is essential for faithful and drift-free image recovery, and it is further illustrated in Fig. 3b and Algorithm 1.

A.3 DNA Editing with Residual Offset

Similarly, during the editing process (as illustrated in Fig. 3c and detailed in Algorithm 2), we incorporate the residual offset back into the latent representations at each timestep. Specifically, we adjust each latent Z_t^{tgt} by adding the pre-computed residual offset Δx_t^{DNA} , yielding a corrected latent Z_t^{*tgt} . This adjustment ensures that the velocity field v_t^{tgt} is computed using a latent aligned with the same noise used in the inversion process, thereby maintaining consistency between the inversion and editing trajectories. This alignment is crucial for preserving the structural integrity of the unedited regions. Without it, the mismatch between inversion and editing latents can lead to misaligned gradient directions, causing semantic drift and unintended changes to the background or identity of the source image. By correcting the editing trajectory with the residual offset, our method maintains a higher fidelity to the original content while allowing more precise and localized editing aligned with the target prompt.

Algorithm 2 Editing Process with Residual Offset

Input: Source image Z_T , Timesteps $\{\sigma_t\}_{t=0}^T$, RF v_θ , Target text ψ^{tgt}
Output: Edited image Z_T^{tgt}
Inversion: See Algorithm 1
Editing Init: $Z_0^{tgt} \leftarrow S_0$
for $t = 0, 1, \dots, T-1$ **do**
 $Z_t^{*tgt} = Z_t^{tgt} + \Delta x_t^{\text{DNA}}$ ▷ Add the residual offset back
 $v_t^{tgt} = v_\theta(Z_t^{*tgt}, \psi^{tgt})$ ▷ Calculate velocity v_t^{tgt} using target text
 $Z_{t+1}^{tgt} = Z_t^{tgt} - v_t^{tgt} \times (\sigma_{t+1} - \sigma_t)$
end for
return Z_T^{tgt}

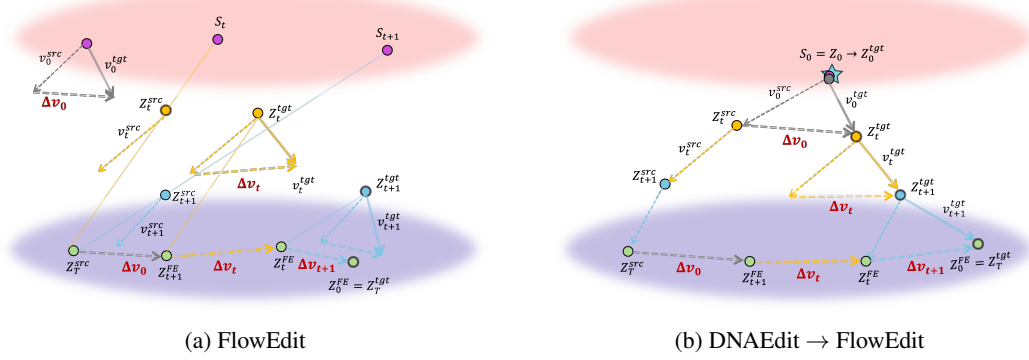


Figure 4: Illustration of the equivalence between FlowEdit and DNAEdit.

B Theoretical Analysis between DNAEdit and FlowEdit

DNAEdit, as illustrated in Fig. 4b, can be viewed as a perfectly aligned, inversion-aware extension of FlowEdit (Fig. 4a). Unlike FlowEdit, which relies on independently sampled Gaussian noise at each timestep, DNAEdit assigns an optimal noise vector to each timestep via direct noise alignment. This alignment ensures the continuity of the target trajectory Z_t^{tgt} across timesteps and eliminates the accumulation of discretization errors that typically arise in FlowEdit.

The overview of FlowEdit is shown in Fig. 4a. It initializes $Z_T^{\text{FE}} = Z_T^{\text{src}}$. For each timestep t , Gaussian noise S_t is randomly sampled, and the source trajectory Z_t^{src} is interpolated between the source image Z_T^{src} and the noise S_t . The target trajectory Z_t^{tgt} is derived using the parallelogram rule:

$$Z_t^{tgt} = Z_{t+1}^{\text{FE}} + Z_t^{\text{src}} - Z_T^{\text{src}}. \quad (6)$$

The velocity difference between the source and target is computed as: $\Delta v_t = v_\theta(Z_t^{tgt}, \psi^{tgt}) - v_\theta(Z_t^{\text{src}}, \psi^{\text{src}})$, and the editing trajectory from the source to target image is updated iteratively: $Z_t^{\text{FE}} = Z_{t+1}^{\text{FE}} + \Delta v_t \times (\sigma_{t+1} - \sigma_t)$.

In contrast, DNAEdit, as shown in Fig. 4b, also constructs a trajectory between the source and target images, but with key improvements. The algorithm initializes $Z_T^{\text{FE}} = Z_T^{\text{src}}$, and for each timestep t , the velocity difference is defined as: $\Delta v_t^{\text{DNA}} = v_\theta(Z_t^{*tgt}, \psi^{tgt}) - v_\theta(Z_t^{*src}, \psi^{\text{src}})$ ($Z_t^{\text{src}*}$ and Z_t^{tgt*} represent the latent variables Z_t^{src} and Z_t^{tgt} with residual offset, respectively; see Fig. 3c). The trajectory from the source to target images is then updated using: $Z_t^{\text{FE}} = Z_{t+1}^{\text{FE}} + \Delta v_t \times (\sigma_{t+1} - \sigma_t)$.

The most critical step is to prove that DNAEdit also satisfies the parallelogram rule (highlighted by the yellow parallelogram in Fig. 5b), as what is done in FlowEdit (yellow parallelogram in Fig. 5a), i.e.,

$$Z_t^{*tgt} = Z_{t+1}^{\text{FE}} + Z_t^{*src} - Z_T^{\text{src}}. \quad (7)$$

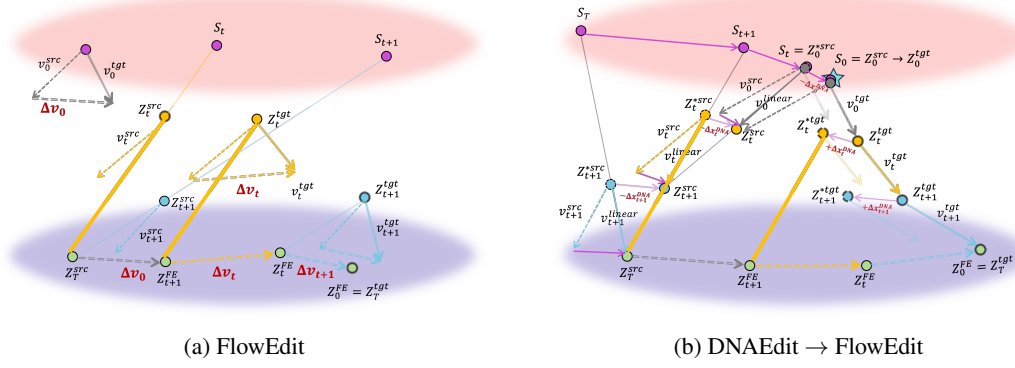


Figure 5: Parallelogram rule in FlowEdit and DNAEdit.

If this rule holds, the only difference between DNAEdit and FlowEdit lies in the selection of Gaussian noise at each timestep. The proof process is as follows:

$$\begin{aligned}
Z_t^{*tgt} &= Z_t^{tgt} + \Delta v_t^{\text{DNA}} \times (\sigma_{t+1} - \sigma_t) \\
&= Z_t^{tgt} + Z_t^{*src} - Z_t^{src} \\
&= Z_t^{tgt} + Z_t^{*src} - Z_T^{src} + Z_T^{src} - Z_t^{src} \\
&= Z_T^{src} + Z_t^{tgt} - Z_t^{src} + Z_t^{*src} - Z_T^{src} \\
&= Z_T^{src} + (Z_0^{tgt} + \sum_{i=0}^{t-1} Z_{i+1}^{tgt} - Z_i^{tgt}) - (Z_0^{src} + \sum_{i=0}^{t-1} Z_{i+1}^{src} - Z_i^{src}) + Z_t^{*src} - Z_T^{src} \\
&= Z_T^{src} + \sum_{i=0}^{t-1} (v_i^{tgt} - v_i^{src}) \times (\sigma_{i+1} - \sigma_i) + Z_t^{*src} - Z_T^{src} \\
&= Z_{t+1}^{\text{FE}} + Z_t^{*src} - Z_T^{src}
\end{aligned} \tag{8}$$

As shown in Fig. 5b, direct noise alignment ensures the relationship: $v_t^{\text{src}} \times (\sigma_{t+1} - \sigma_t) = Z_{t+1}^{\text{src}} - Z_t^{\text{src}}$. However, in FlowEdit, this relationship does not hold because Z_t^{src} and Z_{t+1}^{src} are generated by interpolating the source image with independently sampled noises. As a result, their difference does not necessarily equal the velocity vector, *i.e.*, $v_t^{\text{src}} \times (\sigma_{t+1} - \sigma_t) \neq Z_{t+1}^{\text{src}} - Z_t^{\text{src}}$. This inconsistency in FlowEdit is resolved by the direct noise alignment in DNAEdit, which ensures both the continuity and optimality of the trajectory.

Both DNAEdit and FlowEdit maintain a trajectory from the source to target image $\{Z_t^{\text{FE}}\}_{t=0}^T$. However, the primary difference lies in the continuity of the target trajectory $\{Z_t^{tgt}\}_{t=0}^T$. Specifically, in FlowEdit, the Gaussian noise for each timestep is independently sampled, which prevents the target trajectory from being continuous across adjacent timesteps. In contrast, DNAEdit aligns the noise from S_{t+1} to S_t , ensuring that the target trajectory is continuous across timesteps. This alignment enables DNAEdit to select an optimal noise sample, thereby eliminating discretization errors and producing a smoother trajectory.

C More Experiment Results

C.1 Implementation Details

In our experiments, we utilize the official implementations and default hyperparameters for those comparative methods [6, 9, 2, 1, 11, 3]. The exception is FTEdit [12], which is currently unavailable, and we use the results provided in its paper on the PIE-Bench. For our DNAEdit, the FLUX version is built upon the FLUX-dev [4] model, with the DNA steps set to 28, t_s set to 4 and CFG set to 2.5. In the SD version, we use the SD3.5-medium [7] model, with DNA steps set to 40, t_s set to 13, and CFG set to 3.5. Both versions have the MVG coefficient η set to 0.8 to balance background preservation and target editing. All experiments are conducted on an NVIDIA L40S GPU.




<p>System:</p> <p>Task: Please provide a more detailed description by combining the content of the image and the short prompt for the image-caption pairs I have provided to you, and simultaneously generate the corresponding editing prompts.</p> <p>Guidelines:</p> <ul style="list-style-type: none"> - Please expand the description of the original image focusing on the target that needs editing, and simultaneously generate corresponding editing prompts. Both should be thorough yet concise, under 70 words. - Based on the provided editing instructions, generate new editing commands for the updated caption, requiring that they be as similar as possible in terms of grammar. - The new long editing prompt should only modify the parts in the original editing prompt that are different from the original short prompt. - Format your output as a strings without using the newline character. <p>Example:</p> <pre> input: { original_prompt: "a [dog] is laying down on a white background", editing_prompt: "a [lion] is laying down on a white background" } output: { "new_long_prompt": "A dog with a fluffy coat and short legs is laying down on a pristine white background, showcasing its relaxed posture and soft fur.", "new_long_editing_prompt": "A lion with a fluffy coat and short legs is laying down on a pristine white background, showcasing its relaxed posture and soft fur." } </pre>		
<p>User:</p> 	<p>Original prompt: a yellow bird with a red beak sitting on a branch</p> <p>Editing prompt: a crochet bird with a red beak sitting on a branch</p>	<p>Response:</p> <p>New long prompt: A vibrant yellow bird with a striking red beak is perched gracefully on a slender branch, surrounded by lush green leaves and a textured tree trunk, creating a serene natural setting.</p> <p>New long editing prompt: A vibrant crochet bird with a striking red beak is perched gracefully on a slender branch, surrounded by lush green leaves and a textured tree trunk, creating a serene natural setting.</p>
<p>User:</p> 	<p>Original prompt: wolf howling at the moon with a outline of a wolf</p> <p>Editing prompt: wolf howling at the moon with a outline of a person</p>	<p>Response:</p> <p>New long prompt: A wolf is howling at the moon, silhouetted against a vibrant night sky filled with stars and surrounded by tall pine trees, all within the outline of a wolf's head.</p> <p>New long editing prompt: A wolf is howling at the moon, silhouetted against a vibrant night sky filled with stars and surrounded by tall pine trees, all within the outline of a person's head.</p>
<p>User:</p> 	<p>Original prompt: smoke</p> <p>Editing prompt: fire</p>	<p>Response:</p> <p>New long prompt: Thick plumes of smoke billow from the tall chimneys of an industrial plant, set against a backdrop of a clear blue sky with scattered clouds. The smoke rises high, creating a stark contrast with the surrounding environment, while a wind turbine stands nearby, highlighting the juxtaposition of industrial and renewable energy sources.</p> <p>New long editing prompt: Thick plumes of fire billow from the tall chimneys of an industrial plant, set against a backdrop of a clear blue sky with scattered clouds. The fire rises high, creating a stark contrast with the surrounding environment, while a wind turbine stands nearby, highlighting the juxtaposition of industrial and renewable energy sources.</p>

Figure 6: Detailed prompt for constructing DNA-Bench and samples from DNA-Bench.

C.2 Details of DNA-Bench

We employ the advanced GPT-4o [8] to expand the short prompts in PIE-Bench into longer ones, creating our DNA-Bench. As illustrated in Fig. 6, we design prompts to achieve this expansion. Initially, reference images are input into GPT-4o to parse content and generate descriptions. To prevent hallucinations and ensure the description aligns with the editing intent, we provide the original short prompts, transforming the task into one that expands around these prompts. This method allows for a detailed description of the editing target while offering a comprehensive background description, enhancing alignment between image and text for improved background preservation and editing.

Our strategy effectively expands the original short prompts into longer ones. For example, in the 3rd row of Fig. 6, the original prompt was simply "smoke", which is too short to describe the editing target. In DNA-Bench, the image content is described in detail, aiding in preserving the original image's structure and non-editing areas during editing, while accurately modifying the editing target. We present experimental results and discussions on training-free editing using long texts in both the main text and Fig. 11 in this appendix. Although long texts may pose some inconvenience to humans, the progress made by modern MLLMs allows us to easily integrate them into real-world applications to help expand the initial prompts provided by users into longer prompts that are more conducive to editing. Thus, we believe that DNA-Bench, with its high-quality long text prompts, is important for the future development and evaluation of text-guided editing methods.

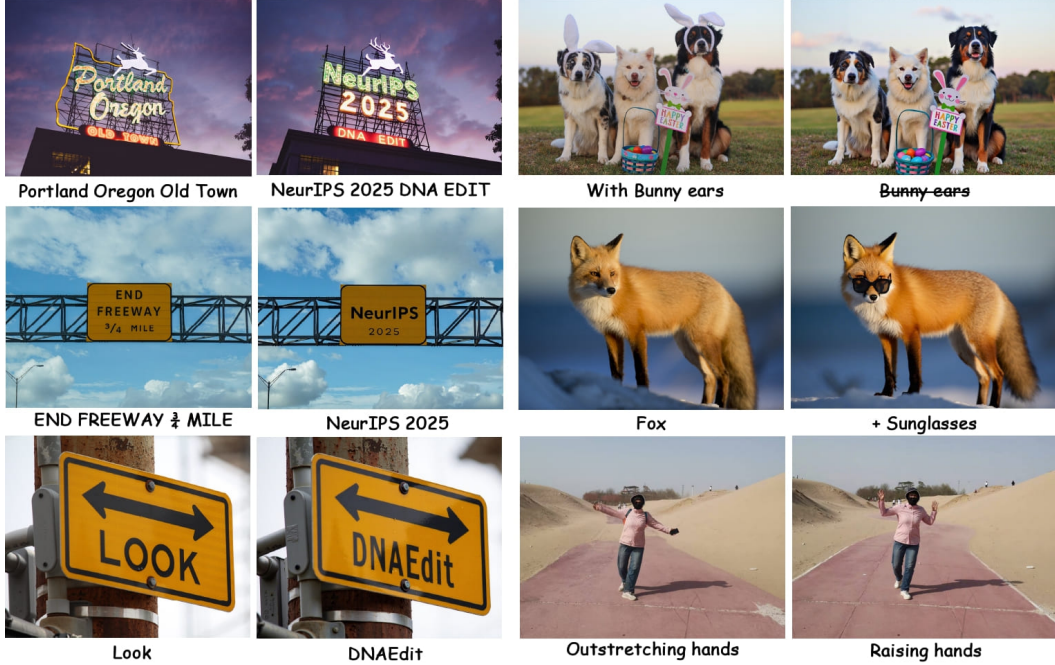


Figure 7: More results of DNAEdit on real images. Original(Left) vs. Edited(Right).

D More Visual Comparisons

D.1 More Visual Results of DNAEdit

In Fig. 7 and Fig. 8, we showcase additional results of DNAEdit applied to high-resolution real-world images. In each image pair, the original is on the left, and the edited result is on the right. DNAEdit demonstrates high-quality editing capabilities on these images. For example, in the first column of Fig. 7, DNAEdit achieves precise text editing, thanks to the powerful text generation capabilities of the T2I model FLUX, while preserving other areas of the image. The second column highlights DNAEdit’s effectiveness in tasks such as addition and deletion. Additionally, in the 3rd row of the right column, DNAEdit successfully changes a human pose from "outstretching hands" to "raising hands," illustrating its capability in non-rigid editing tasks. In Fig. 8, we present more object-level editing results, including changes in object type and material. For instance, in the 2nd column, a lighthouse is transformed into an iron tower, with details like line-shaped clouds in the sky perfectly preserved. The last row demonstrates the effect of changing an object’s material, where the original material is completely altered, yet the object’s original features, such as posture, are retained.

D.2 Comparisons on PIE-Bench

We present additional visual comparisons in Fig. 9, where the left side shows results from methods based on the FLUX model, and the right side displays results from methods based on the SD series. Our approach consistently outperforms other methods across various editing tasks. For instance, in the non-rigid editing task on the left side, row 7, FireFlow fails to make the dog sit, while FlowEdit succeeds but significantly alters the dog’s appearance. In contrast, our method preserves the dog’s original features while successfully changing its posture from standing to sitting. Additionally, as can be seen in row 8 on the left side, our method effectively tackles the challenging task of changing large backgrounds, maintaining the woman’s appearance almost unchanged while transforming the background from a street to a forest. Methods based on SD also achieve similar results. On the right side, row 8, our method successfully stylizes the entire image while preserving the characteristics of the original elements, such as the child’s actions and attire, aligning more closely with our intent.



Figure 8: More results of DNAEdit on real images. Original(Left) vs. Edited(Right).

D.3 Comparisons on DNA-Bench

In Fig. 10, we additionally present comparative results under long text descriptions from the DNA-Bench. With the introduction of long text, RF-based editing methods show some improvements, such as more accurate backgrounds. However, our method still maintains advantages. For example, in the 4th row on the left, FireFlow alters the entire image structure. Although FlowEdit appears to preserve the structure, a comparison with our method reveals that both the person and the background undergo significant changes. Similarly, in the 5th row, our method better preserves the background (such as the tree trunk) and the sunglasses on the person’s face, while successfully changing the book into a laptop.

E More Ablation Studies

E.1 Ablation on Effect of Long-text Prompt in DNA-Bench

In Fig. 11, we present a comparison of various methods on PIE-Bench and DNA-Bench to demonstrate the impact of long text on image editing. As discussed in Sec 4.2 of main paper, a longer source prompt can provide a more detailed description of the image, resulting in noise that is more aligned with the source prompt. During editing, since the target prompt is similar to the source prompt, the non-edit regions remain unchanged. For example, in the 1st row, 3rd column of Fig. 11, under a long text prompt, the description "cobblestone path" leads to noticeable changes in the ground in FireFlow’s editing results, making it more consistent with the original image. Our DNAEdit method also benefits from this; under a short text prompt, the paw area transforms into a larger flower, whereas under a long text prompt, both the paw and flower are more consistent with the original image. This effect is even more pronounced in the third row, where both FireFlow and DNAEdit show significant changes in the image background under short text prompts. However, under long text prompts, thanks to the detailed background description, the edited results better match the original scene. In addition, long text prompts enable for more precise editing targets. In the second row, the editing goal is to replace the woman with a storm-trooper. Under the limited expression of a short text prompt, the model mistakenly applies the edit to the shirt’s pattern. With a long text prompt, which provides more accurate alignment between the scene and text, the edit is successfully applied to the woman. Combined with the results shown in the main paper, we can see that long text prompts in DNA-Bench offer significant benefits for the most recent RF-based editing models.

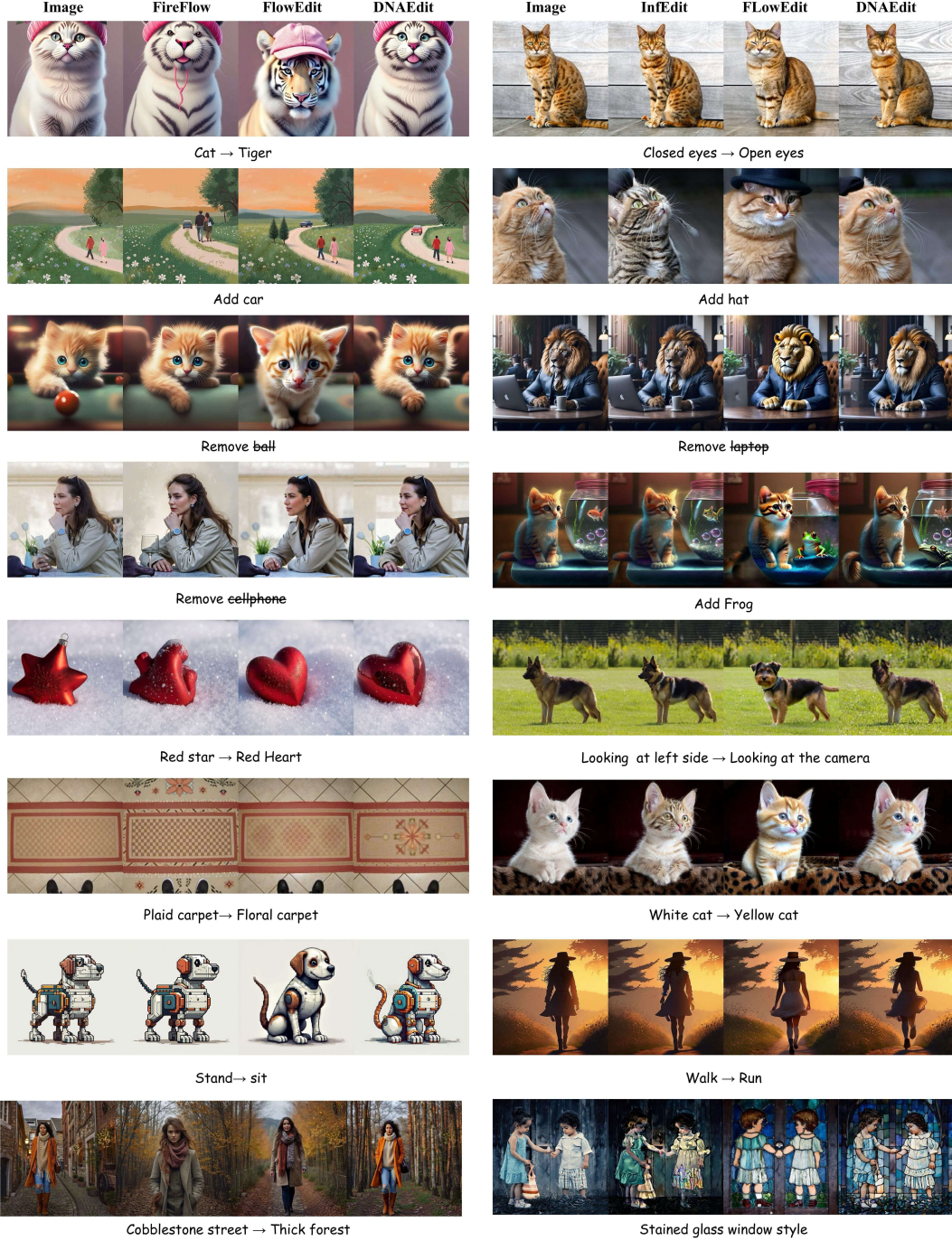


Figure 9: More visual comparisons on PIE-Bench. Left: FLUX-based, Right: SD-based.

E.2 Ablation on Effect of MVG Coefficient η

In Fig. 12, we demonstrate the impact of different MVG coefficients on editing results. Fig. 12 clearly shows that MVG can effectively ensure the faithfulness to the reference image. For instance, in the third row, when the MVG is set to 1.0, the overall layout of the image changes significantly. As the MVG decreases, the structure becomes more similar to the original image, with flowers added to the grass while maintaining structural consistency at MVG values of 0.8 and 0.7. However, an excessively large MVG can lead to editing failures. For example, in the 1st row, with an MVG of

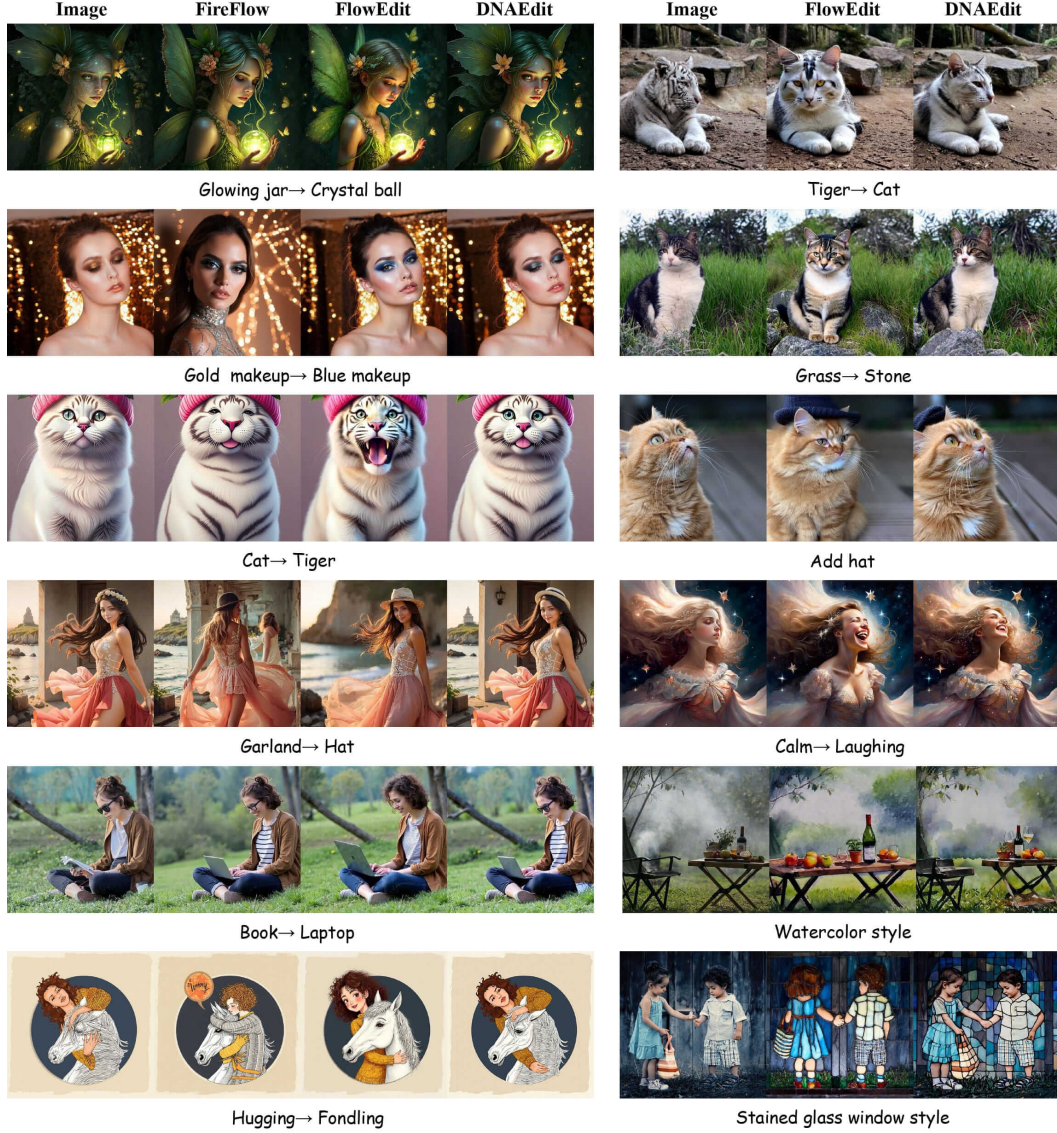


Figure 10: More visual comparisons on DNA-Bench. Left: FLUX-based, Right: SD-based.

Table 1: Ablation on MVG coefficient η .

Exp.	η	Struct. \downarrow	PSNR \uparrow	LPIPS \downarrow	MSE \downarrow	SSIM \uparrow	CLIP-W \uparrow	CLIP-E \uparrow
①	1.0	33.98	21.97	149.84	95.19	79.32	26.40	23.23
②	0.9	24.52	23.64	118.20	67.05	83.06	26.16	22.98
③	0.8	18.87	24.99	95.06	50.45	85.71	25.79	22.87
④	0.7	15.32	26.06	79.43	39.20	87.58	25.39	22.06

0.7, the dog is not edited to sit, whereas results with MVG greater than 0.8 successfully achieve this. Moreover, compared to MVG values of 0.9 and 1.0, the result at 0.8 more closely resembles the original appearance of the dog, aligning with our editing intent.

Quantitatively, Table 1 reflects the same trend as shown in the visual results. $\eta = 1.0$ achieves the strongest CLIP scores but over-edits structure/background. Lowering $\eta = 1.0$ to 0.9–0.8 substantially improves structural and background metrics with only a slight drop in CLIP, striking a good balance. $\eta = 0.7$ further boosts preservation but starts to weaken the edit semantics. In summary, η in the



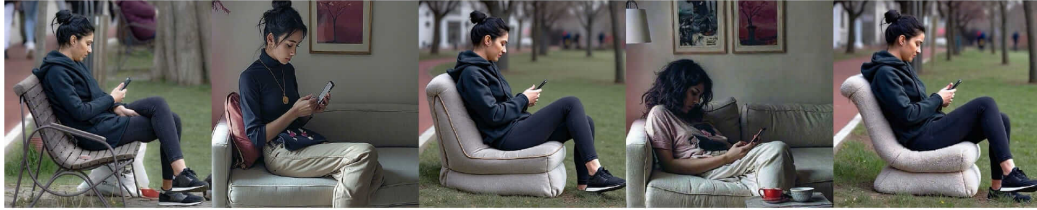
PIE-Bench:
blue light, a black and white **cat-dog** is playing with a flower

DNA-Bench:
Under a soft blue light, a playful black and white **cat dog** with bright eyes and fluffy fur is gently pawing at a delicate flower on a cobblestone path, creating a charming and serene scene.



PIE-Bench:
a **woman storm-trooper** with blue hair wearing a shirt

DNA-Bench:
A **woman storm-trooper** with vibrant blue hair cascading over her shoulders is wearing a casual white shirt, set against a lush green forest background.



PIE-Bench:
a woman sitting on a **bench sofa** using her cell phone

DNA-Bench:
A woman dressed in a dark hoodie and leggings is sitting on a metal and **wooden-bench-sofa** in a park, engrossed in her cell phone. The bench is placed on a grassy area near a walking path, with trees and other people in the background.



PIE-Bench:
a **rabbit cat** is sitting in a pile of colorful eggs

DNA-Bench:
A **rabbit-cat** with soft, gray fur and long ears is sitting amidst a vibrant pile of colorful eggs, surrounded by lush greenery, creating a playful and festive atmosphere.

Figure 11: Visual comparison between results from PIE-Bench and DNA-Bench.

range 0.9–0.8 provides the best trade-off between edit fidelity and background preservation. This experiment demonstrates that an appropriate MVG can effectively preserve non-editing areas without affecting them. Additionally, by adjusting the MVG, users can control the intensity of the edits according to their needs, achieving more flexible editing results.

E.3 Ablation on Initialization Strategy

We compare our default DNA pipeline initialized with randomly sampled Gaussian noise against a variant initialized with FireFlow’s inverted noise. As shown in Table 2, the two initializations yield

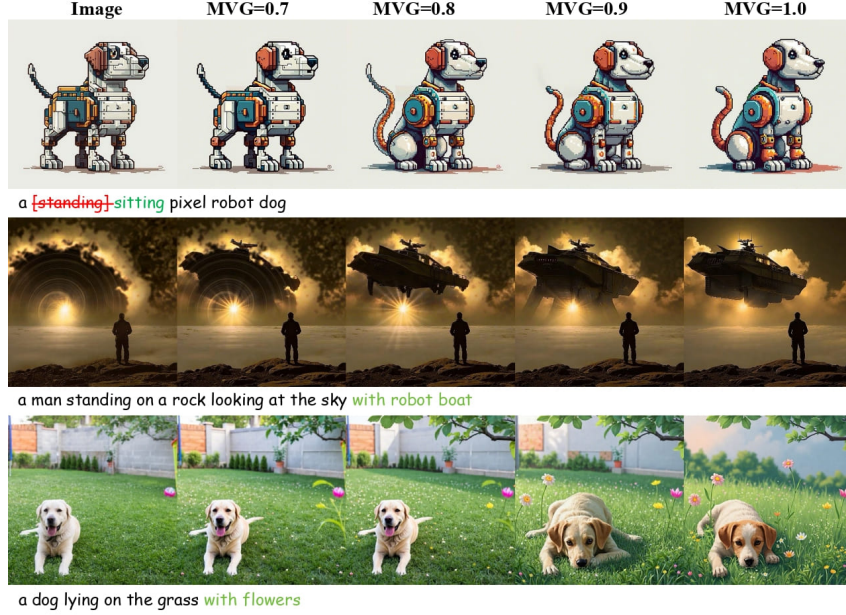


Figure 12: Comparison between different MVG coefficients η .

Table 2: Ablation on initialization strategy: FireFlow inversion-based noise vs. random Gaussian sampling.

Exp.	Structure Dist. ↓	Background Preservation				CLIP Similarity ↑	
		PSNR ↑	LPIPS ↓	MSE ↓	SSIM ↑	Whole ↑	Edited ↑
Inversion Noise Initialization	19.73	24.95	94.47	50.47	85.78	25.77	22.39
Random Gaussian Sampling	18.87	24.99	95.06	50.45	85.71	25.79	22.87
Δ (Random – Inversion)	-0.86	+0.04	+0.59	-0.02	-0.07	+0.02	+0.48

nearly identical performance across structural consistency, background preservation, and semantic alignment. Random initialization is slightly better in Structure Distance (18.87 vs. 19.73), while PSNR and SSIM are essentially on par (24.99 vs. 24.95; 85.71 vs. 85.78). CLIP similarity exhibits only marginal gains over the random variant on both the whole image and edited regions (25.79 vs. 25.77; 22.87 vs. 22.39). However, the inversion-based initialization incurs approximately 50% additional computational overhead. These results indicate that DNAEdit is insensitive to the randomness of the initial noise—any sampled Gaussian noise can be optimized to the desired state—thereby justifying random Gaussian sampling as our default initialization.

F DNAEdit for Video Editing

In Fig. 13, we present the results of DNAEdit applied to more challenging videos editing task [5]. We conduct experiments using the Rectified Flow-based text-to-video (T2V) model, Wan2.1 [10]. Since DNAEdit is a model-agnostic algorithm, we achieve impressive editing effects **without any modifications to the model**. As illustrated, DNAEdit accurately transforms the target, such as changing a bear into a panda, while preserving the original structure and time consistency. This demonstrates DNAEdit’s strong potential for application in various existing and future flow-based generative models, enabling more valuable applications.

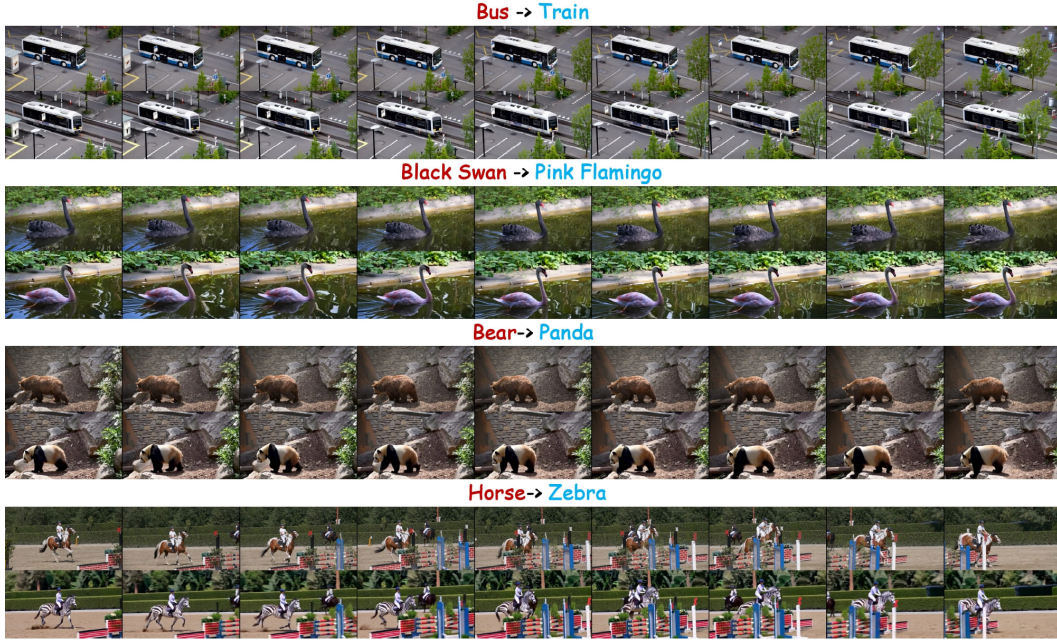


Figure 13: Visual results of DNAEdit for video editing task.

G Broader Impacts

Our proposed image editing method has several potential societal impacts, both positive and negative. On the positive side, this method can enhance creative industries by providing artists and designers with powerful tools for content creation and modification. However, there are potential negative impacts to consider. The technology could be misused for creating misleading or harmful content, which could have significant implications for privacy and security. To mitigate these risks, we suggest implementing mechanisms for monitoring and controlling the use of the technology, such as gated releases, and developing tools to detect and counteract malicious uses.

References

- [1] Yingying Deng, Xiangyu He, Changwang Mei, Peisong Wang, and Fan Tang. Fireflow: Fast inversion of rectified flow for image semantic editing. *arXiv preprint arXiv:2412.07517*, 2024.
- [2] Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. Direct inversion: Boosting diffusion-based editing with 3 lines of code. *arXiv preprint arXiv:2310.01506*, 2023.
- [3] Vladimir Kulikov, Matan Kleiner, Inbar Huberman-Spiegelglas, and Tomer Michaeli. Flowedit: Inversion-free text-based editing using pre-trained flow models. *arXiv preprint arXiv:2412.08629*, 2024.
- [4] Black Forest Labs. Official weights of FLUX.1 dev. <https://huggingface.co/black-forest-labs/FLUX.1-dev>, 2024. Accessed: 2024-11-14.
- [5] Minghan Li, Chenxi Xie, Yichen Wu, Lei Zhang, and Mengyu Wang. Five: A fine-grained video editing benchmark for evaluating emerging diffusion and rectified flow models. *arXiv preprint arXiv:2503.13684*, 2025.
- [6] Litu Rout, Yujia Chen, Nataniel Ruiz, Constantine Caramanis, Sanjay Shakkottai, and Wen-Sheng Chu. Semantic image inversion and editing using rectified stochastic differential equations. *arXiv preprint arXiv:2410.10792*, 2024.
- [7] StabilityAI. Official weights of SD3.5 medium. <https://huggingface.co/stabilityai/stable-diffusion-3.5-medium>, 2024. Accessed: 2024-11-14.
- [8] OpenAI Team. Gpt-4o system card, 2024.
- [9] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1921–1930, 2023.
- [10] Team Wan. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.
- [11] 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. *arXiv preprint arXiv:2411.04746*, 2024.
- [12] Pengcheng Xu, Boyuan Jiang, Xiaobin Hu, Donghao Luo, Qingdong He, Jiangning Zhang, Chengjie Wang, Yunsheng Wu, Charles Ling, and Boyu Wang. Unveil inversion and invariance in flow transformer for versatile image editing. *arXiv preprint arXiv:2411.15843*, 2024.