PEdit: Pareto-Guided Image Editing via Dynamic Latent Trajectory Control

Supplementary Material

In this supplementary material, we provide,

A. Experimental Details

In this section, additional experimental details of the proposed method are provided. First, the analysis that motivates the approach is described (Section A.1), focusing on the investigation of editing behaviors within the DiT architecture. Next, the methodology of the zero-guiding experiment is outlined (Section A.2), followed by the corresponding experimental results on the Kontext model (Section A.3) and Qwen-ImageEdit (Section A.4).

A.1. Motivation

Fig. A shows that applying a fixed, unified condition strength in DiT-based architectures leads to imbalanced attention between text and source-image conditions, resulting in biased information flow. Previous T2I studies have attempted to regulate such imbalance through explicit attention control mechanisms [2, 5, 12], which amplify token-level attention scores to strengthen semantic transmission and improve text–image alignment. However, image editing fundamentally differs from T2I, it requires not only semantic fidelity to the text condition but also structural preservation of the source image. Balanced semantic transmission can only be achieved when the ratio between TCA and ICA remains near an optimal equilibrium.

As illustrated in Fig. A (right), structural failure cases (blue) appear similar to optimal cases (green) when viewed in the attention space. However, despite this similarity, the model ultimately fails to preserve structural information. This observation suggests that attention alone is insufficient to determine whether structural cues are properly conveyed. This limitation contrasts with UNet–based diffusion models, where image conditions are concatenated along spatial channels, allowing explicit propagation of spatial features. In DiT, however, both text and image conditions are sequencially concatenated, enhancing semantic correspondence, but weakening spatial fidelity, leading to structural distortion (see Fig. A, left). This implies two key requirements: (1) maintaining an optimal balance between TCA and ICA to prevent bias toward either condition, and (2) introducing an auxiliary mechanism beyond cross-attention to preserve spatial information. To this end, SSNR is introduced to quantify the degree of structural preservation.

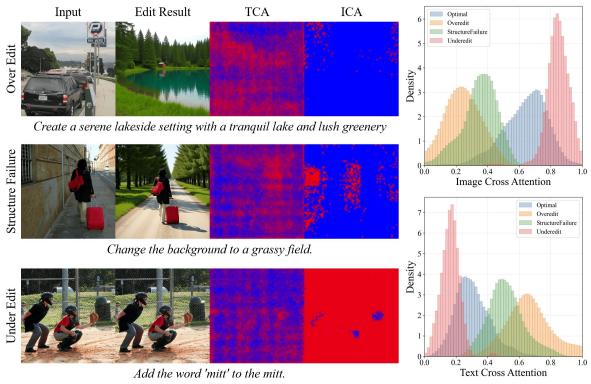


Figure A. **Visualization and statistical results of naive editing and corresponding attention distributions. Left:** Qualitative examples show that transformer based architectures fail to properly edit image. The model either ignores the given input conditions or responds solely to the semantics of the editing prompt, neglecting spatial structural information. This failure is evident in the contrasting cross-attention maps. In the first denoising step, samples with unsuccessful editing exhibit biased attention-score distributions, indicating an imbalance between the two conditioning sources. **Right:** The statistical plots show the distributions of TCA ICA across different editing cases. Over-edited cases tend to have higher TCA scores, implying excessive reliance on textual guidance, whereas under-edited cases exhibit higher ICA values, reflecting stronger dependence on visual information from the source image. In structural failure cases, the distributions display intermediate characteristics between Optimal and Overedit, with degraded spatial consistency.

A.2. Zeroing Experiment

A zeroing experiment is conducted using two DiT-based models, Kontext and Qwen-ImageEdit. As described in Section 3.3, this ablation study sequentially zeros out the text or image condition within each Transformer block. Using the same initial noise and denoising steps, the resulting images are compared to analyze how each condition contributes to text-guided edit and source-image preservation.

For FLUX, 28 denoising steps and 57 Transformer blocks are used, while Qwen-ImageEdit uses 50 steps across 60 blocks. Experiments are conducted on the Emu-

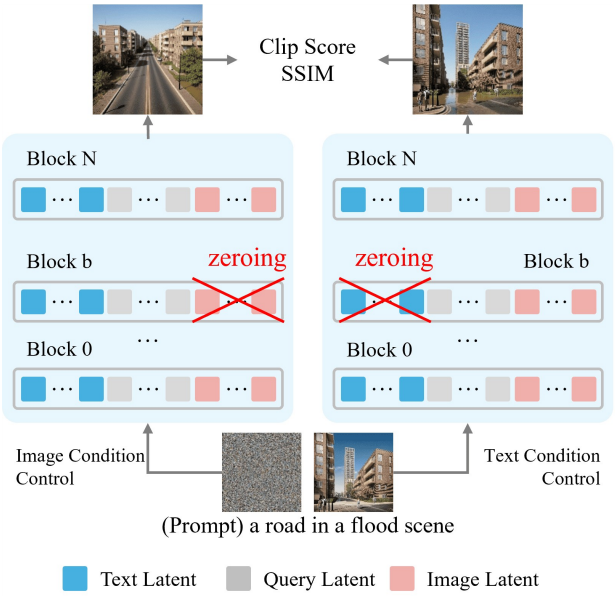


Figure B. **Visualization of zeroing experiment setup.** The DiT model is composed of transformer blocks [10], where text and source image conditions exist independently in the latent space (blue tokens represent text latents, and pink tokens represent image latents). This separation enables independent analysis of their functional contributions. To examine the behavior of each block, the corresponding condition is zeroed out and the editing outcomes are evaluated using CLIP-Dir and SSIM [40].

Edit validation set, with results averaged over inference outputs. CLIP-Dir and SSIM are used to evaluate text alignment and structural preservation, respectively, and only cases where one metric improves are considered for clarity.

It is observed that zeroing a condition at certain blocks can produce counterintuitive effects: removing the image condition may further emphasize the source image, while removing the text condition can strengthen textual emphasis. This indicates block-dependent behavior, where suppressing a condition can instead amplify its effect. These findings suggest that block-wise condition scaling enables fine-grained control over how different layers influence the final edit.

A.3. Kontext Blockwise Function

Zeroing analysis on Kontext reveals two distinct functional tendencies among Transformer blocks. Zeroing the image condition at certain blocks leads to higher CLIP-Dir but lower SSIM, indicating edit-oriented behavior (blocks 0, 1, 2, 13, 36, 39, 43, 47, 51). In contrast, zeroing the text condition increases SSIM, reflecting source-preserving behavior (blocks 1, 3, 4, 5, 7, 8, 11, 12, 14, 16, 17, 18, 19, 20, 22, 23, 25, 26, 28, 31, 35, 39, 40, 41, 44, 46, 47, 49, 51, 54). Additionally, high CLIP-Dir scores are observed when zeroing the image condition at blocks 29, 31, 32, 37, and the text condition at blocks 9, 24, 27, 32, 37, 38, 43, 45, 48, 50, 52.

Fig.C illustrate the qualitative results of the zeroing experiments. In each grid, the left set corresponds to the image-zeroing results, while the right set shows the text-zeroing results. For each set, the first image (highlighted with a red border) represents the original input, and the second image (highlighted with a blue border) depicts the editing result under the baseline (naive inference). Images highlighted with a yellow border indicate blocks where zeroing the corresponding condition leads to better preservation of the source image. Conversely, images highlighted with a green border represent blocks where zeroing the condition results in stronger editing behavior.

A.4. Qwen-ImageEdit Blockwise Function

The blockwise behavior of Qwen-ImageEdit is also analyzed. Compared to Kontext, less pronounced changes are observed under layer-wise zeroing, and the separation between source preservation and editing behavior is less distinct. While Kontext injects source and text information through clearly separated pathways, Qwen-ImageEdit encodes source information through both the image condition and the textual prompt, leading to more entangled roles. Using the same zeroing protocol, layers related to source preservation and editing are identified. Zeroing the image condition at block 19, and the text condition at blocks 4, 19, 31, 44, and 50 increases SSIM. In contrast, zeroing the image condition at blocks 26 and 57, and the text condition at blocks 3 and 56 yields higher CLIP-Dir, indicating edit-oriented behavior. Fig. D shows the results.

A.5. PEdit Algorithm Details.

In PEdit, Stage 1 performs an optimization-based search to identify the optimal condition scales over n iterations, with early stopping applied to reduce computation. The process terminates if the loss falls below θ of the initial value. Empirically, $\theta = 0.7$ and $n = 2$.

In Stage 2, the method adaptively updates the condition combination according to the current editing state, guided by the auxiliary condition of M_e and M_s. However, when the update trajectory in Stage 2 deviates from the Pareto front, the system switches back to Stage 1 and re-performs condition optimization.

Furthermore, if the current state fails to satisfy Pareto dominance with respect to M_e and M_s for m steps consecutively, specifically when the SSNR ordering deviate from $\mathrm{SSNR}(z_s) > \mathrm{SSNR}(z_x) > \mathrm{SSNR}(z_e)$ and $\mathrm{TCR}(z_e) > \mathrm{TCR}(z_x) > \mathrm{TCR}(z_s)$, it returns to Stage 1. The value of m is set to 3 empirically.

Algorithm 1 Stage 1: PEdit Optimization Algorithm

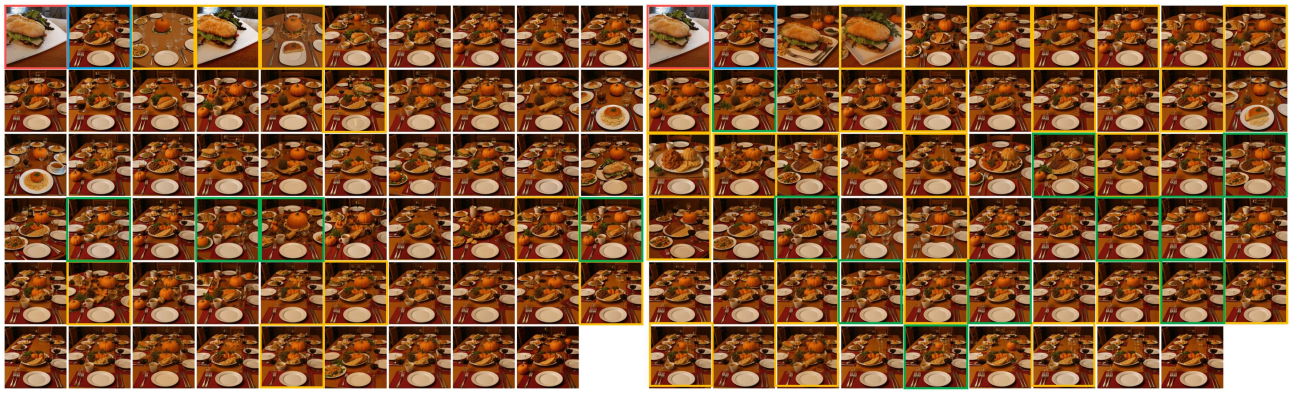
Require: Model M with initial condition configuration, maximum iterations N
Ensure: Optimized model $M^\star$ with controlled condition configuration

Initialize parameters of M and optimizer
Let $M_\mathrm{e}, M_\mathrm{s} \in \mathcal{C}$ be two reference configurations
for $n = 1, \ldots, N$ **do**
  Compute $\big(\mathrm{SSNR}(z_\mathrm{x}), \mathrm{TCR}(z_\mathrm{x})\big)$ ▷ see Eq. 6 and Eq. 9
  Compute $\big(\mathrm{SSNR}(z_\mathrm{e}), \mathrm{TCR}(z_\mathrm{e})\big)$
  Compute $\big(\mathrm{SSNR}(z_\mathrm{s}), \mathrm{TCR}(z_\mathrm{s})\big)$
  if $\mathrm{MSE}(\mathrm{SSNR}(z_\mathrm{s}), \mathrm{SSNR}(z_\mathrm{x})) < \mathrm{MSE}(\mathrm{SSNR}(z_\mathrm{e}), \mathrm{SSNR}(z_\mathrm{x}))$ **then**
    $\mathcal{L}_{\text{align-target}} \leftarrow \mathrm{MSE}(M_\mathrm{e}, M_\mathrm{x})$
  else
    $\mathcal{L}_{\text{align-target}} \leftarrow \mathrm{MSE}(M_\mathrm{s}, M_\mathrm{x})$
  end if
  $\mathrm{SSNR}(z_\text{target}) \leftarrow \dfrac{\mathrm{SSNR}(z_\mathrm{s}) + \mathrm{SSNR}(z_\mathrm{e})}{2}$
  $\mathrm{TCR}(z_\text{target}) \leftarrow \dfrac{\mathrm{TCR}(z_\mathrm{s}) + \mathrm{TCR}(z_\mathrm{e})}{2}$
  $\mathcal{L}_\text{ssnr} \leftarrow \mathrm{MSE}(\mathrm{SSNR}(z_\mathrm{x}), \mathrm{SSNR}(z_\text{target}))$
  $\mathcal{L}_\text{tcr} \leftarrow \mathrm{MSE}(\mathrm{TCR}(z_\mathrm{x}), \mathrm{TCR}(z_\text{target}))$
  $\mathcal{L}_\text{align} \leftarrow \mathcal{L}_\text{align-target}$ ▷ see Eq. 12
  $\mathcal{L}_\text{total} \leftarrow \alpha_\text{ssnr}\,\mathcal{L}_\text{ssnr} + \alpha_\text{tcr}\,\mathcal{L}_\text{tcr} + \alpha_\text{align}\,\mathcal{L}_\text{align}$ ▷ see Eq. 13
  Update M by taking an optimizer step with respect to $\mathcal{L}_\text{total}$
end for
return $M^\star$

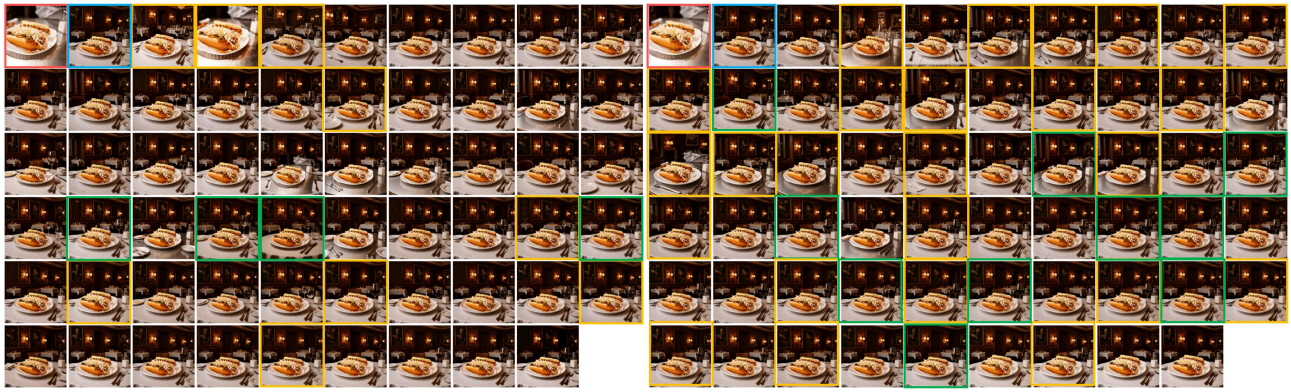
Algorithm 2 Stage 2: Algorithm for Maintaining the Pareto Front

Require: Model M, reference configurations $M_\mathrm{e}, M_\mathrm{s} \in \mathcal{C}$ from Stage 1
Ensure: Optimized model $M^\star$ with controlled condition configuration

Compute $\big(\mathrm{SSNR}(z_\mathrm{x}), \mathrm{TCR}(z_\mathrm{x})\big)$ from M
Compute $\big(\mathrm{SSNR}(z_\mathrm{s}), \mathrm{TCR}(z_\mathrm{s})\big)$ from M_s
Compute $\big(\mathrm{SSNR}(z_\mathrm{e}), \mathrm{TCR}(z_\mathrm{e})\big)$ from M_e
`Aligned_SSNR` $\leftarrow$ `False`
`Aligned_TCR` $\leftarrow$ `False`
if $\mathrm{SSNR}(z_\mathrm{s}) > \mathrm{SSNR}(z_\mathrm{x}) > \mathrm{SSNR}(z_\mathrm{e})$ **then**
  `Aligned_SSNR` $\leftarrow$ `True`
end if
if $\mathrm{TCR}(z_\mathrm{s}) < \mathrm{TCR}(z_\mathrm{x}) < \mathrm{TCR}(z_\mathrm{e})$ **then**
  `Aligned_TCR` $\leftarrow$ `True`
end if
if `Aligned_SSNR` **and not** `Aligned_TCR` **then**
  $M \leftarrow \frac{M+M_\mathrm{s}}{2}$
else if not `Aligned_SSNR` **and** `Aligned_TCR` **then**
  $M \leftarrow \frac{M+M_\mathrm{e}}{2}$
else if not `Aligned_SSNR` **and not** `Aligned_TCR` **then**
  Go to Stage 1
end if



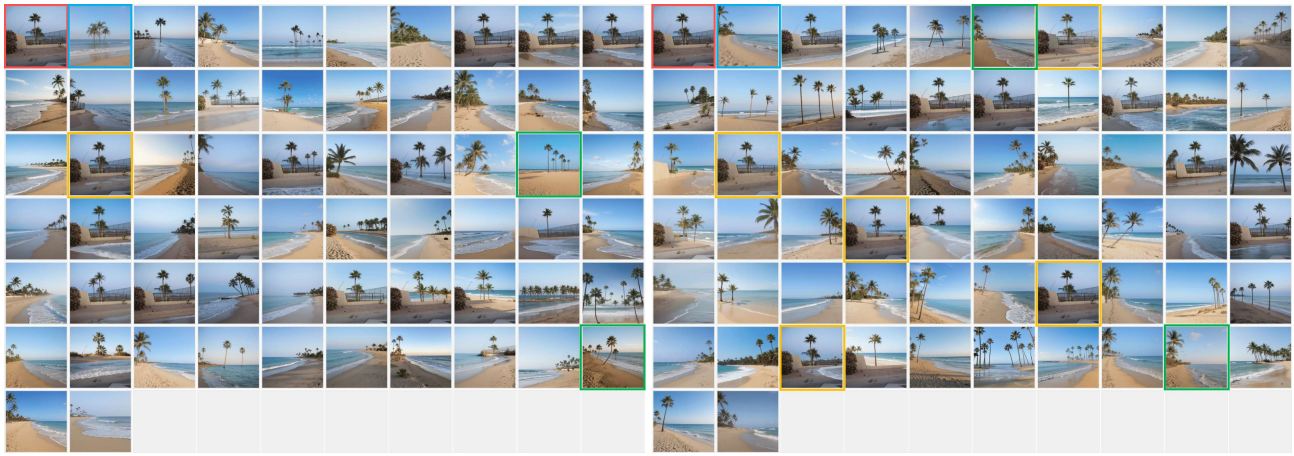
(a) Text editing prompt: *"change the background to a line of kids waiting for lunch."*



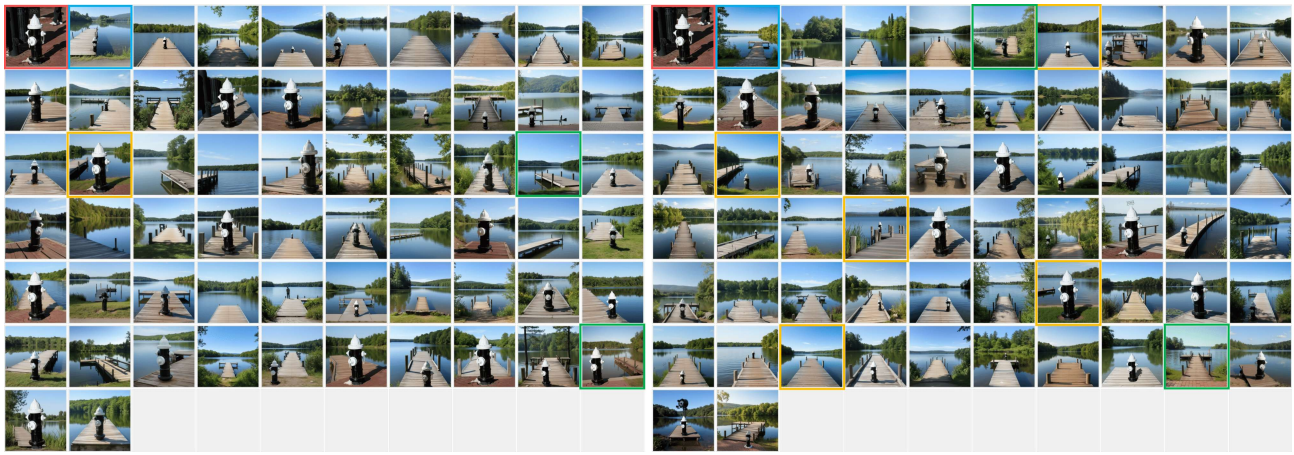
(b) Text editing prompt: *"make the photo seem like it was taken in a fancy restaurant."*

(c) Text editing prompt: *"change the background so that there is a hillside."*

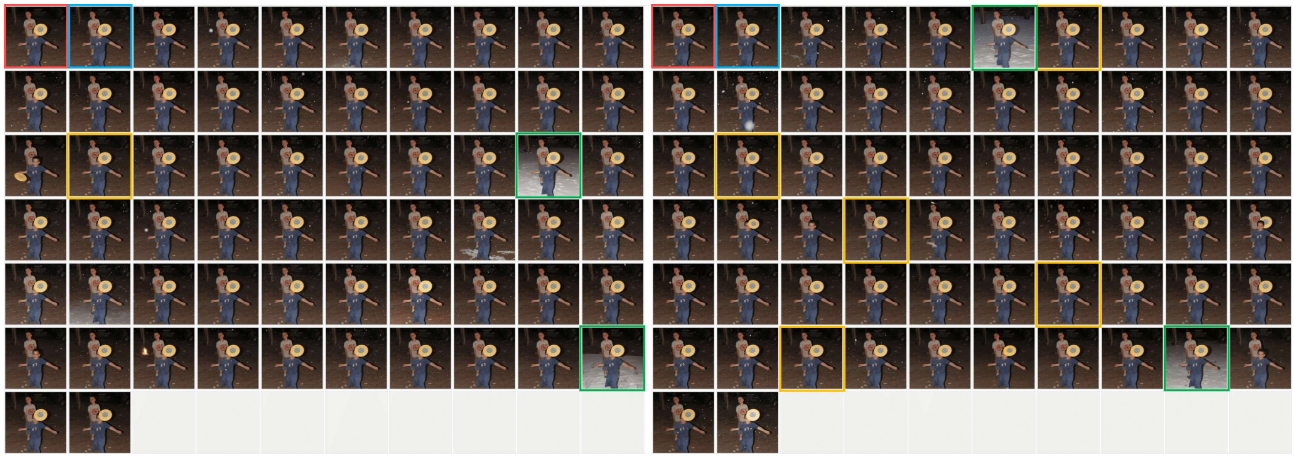
Figure C. Qualitative examples of blockwise zeroing analysis in Kontext.



(a) Text editing prompt: *"Create a serene beach scene with gentle waves lapping against the shore, palm trees swaying in the breeze, and a clear blue sky overhead."*



(b) Text editing prompt: *"Create a serene lakeside setting with a tranquil lake, lush greenery, and a quaint wooden cabin nestled among the trees."*



(c) Text editing prompt: *"Convert it into ink painting."*

Figure D. Qualitative examples of blockwise zeroing analysis in Qwen-ImageEdit.

B. Additional Experiments

In this section, additional experimental details and results that could not be included in the main paper due to page limitations are provided.

B.1. Edit Optimization Stage

Data	Weight		CLIP Dir	CLIP Image	CLIP Text	L2
	Pareto	Align				
Synthetic (HQ-Edit)	1	1	0.051	0.970	0.233	0.007
	1	300	0.069	0.979	0.243	0.006
	1	500	0.050	0.981	0.231	0.005
	300	1	0.041	0.972	0.228	0.012
	500	1	0.047	0.945	0.232	0.020
Real (Emu-Edit)	1	1	0.103	0.919	0.285	0.067
	1	300	0.084	0.921	0.277	0.065
	1	500	0.097	0.924	0.278	0.054
	300	1	0.093	0.917	0.278	0.068
	500	1	0.106	0.940	0.283	0.047

Table A. **Experiment on Pareto loss and align loss weights.** The colored rows indicate the weight actually used in our main experiments. The optimal weights differ between synthetic (HQ-Edit) and real (Emu-Edit) data.

Stage 1 is an optimization-based stage. Suitable loss weights for the Pareto losses, $\mathcal{L}$SSNR and $\mathcal{L}$TCR, and the alignment loss $\mathcal{L}_{\text{Align}}$ are determined by varying their relative ratios and evaluating the results using CLIP-Dir [3], CLIP-I, CLIP-T, and L2 loss for non-edited region preservation. Following [12], a wide range of weight ratios is explored.

Five configurations are evaluated, as summarized in Tab. A, and the optimal ratio is observed to depend on the dataset. Synthetic data (HQ-Edit) and real data (Emu-Edit) require different loss weights, which is attributed to differences in the model's initial behavior.

When the model is not strongly biased, increasing $\mathcal{L}_{\text{Align}}$ helps correct small deviations and leads to faster, stable convergence. This is typically observed in synthetic data, where severe failure modes are less frequent. In contrast, real data often exhibits stronger bias toward either the source image or the editing prompt. In such cases, increasing $\mathcal{L}_{\text{Align}}$ alone is insufficient, and larger weights on $\mathcal{L}_{\text{SSNR}}$ and $\mathcal{L}_{\text{TCR}}$ are required to guide optimization toward a balanced region.

Overall, the optimal loss-weight ratio in Stage 1 depends on the model's current state, and properly balancing the Pareto and alignment losses enables stable progression toward well-balanced edits.

B.2. Comparison with Previous Method

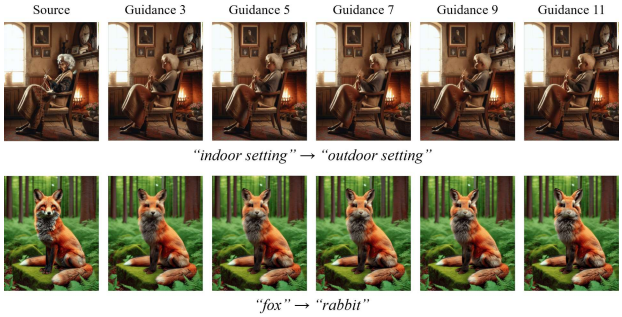


Figure E. **Only initial-noise optimization result.** Given a source image and an editing prompt, text-guidance scaling is applied to TiNO-Edit [37], which optimizes only the initial noise and the starting timestep. Despite adjusting the guidance strength, the edited outputs fail to align with the target semantics. In the global editing case (first row), where an indoor scene should transform into an outdoor environment, almost no meaningful edit is produced while facial structures are severely degraded. In the local editing example (second row), only faint hints of rabbit-like ears are observed, without successfully generating an actual rabbit appearance. These observations indicate that optimizing only the initial timestep is insufficient for controlling the editing trajectory throughout the denoising process, resulting in unsuccessful and unstable edits.

Within sample-wise optimization methods for image editing, TiNO-Edit [37] is closely related, as it also performs per-sample optimization to determine effective input conditions. However, while our method optimizes the scaling of internal condition pathways, TiNO-Edit adjusts only the initial noise and diffusion timesteps.

This design introduces a limitation for text-driven editing, as the optimization is confined to early denoising steps and does not account for the full editing dynamics across later stages of the diffusion process.

As shown in Fig. E, optimizing only early steps fails to achieve meaningful text–image alignment. In one case, the model ignores the instruction to change the scene to an outdoor environment, and in another, the requested transformation from a fox to a rabbit is only weakly reflected and remains semantically inconsistent. Increasing guidance strength yields only marginal improvement, indicating that editing behavior cannot be effectively controlled by manipulating initial noise alone, but requires control throughout the entire diffusion trajectory.

C. Qualitative Results

Additional qualitative results of the proposed method are provided in Fig. F to Fig. I.

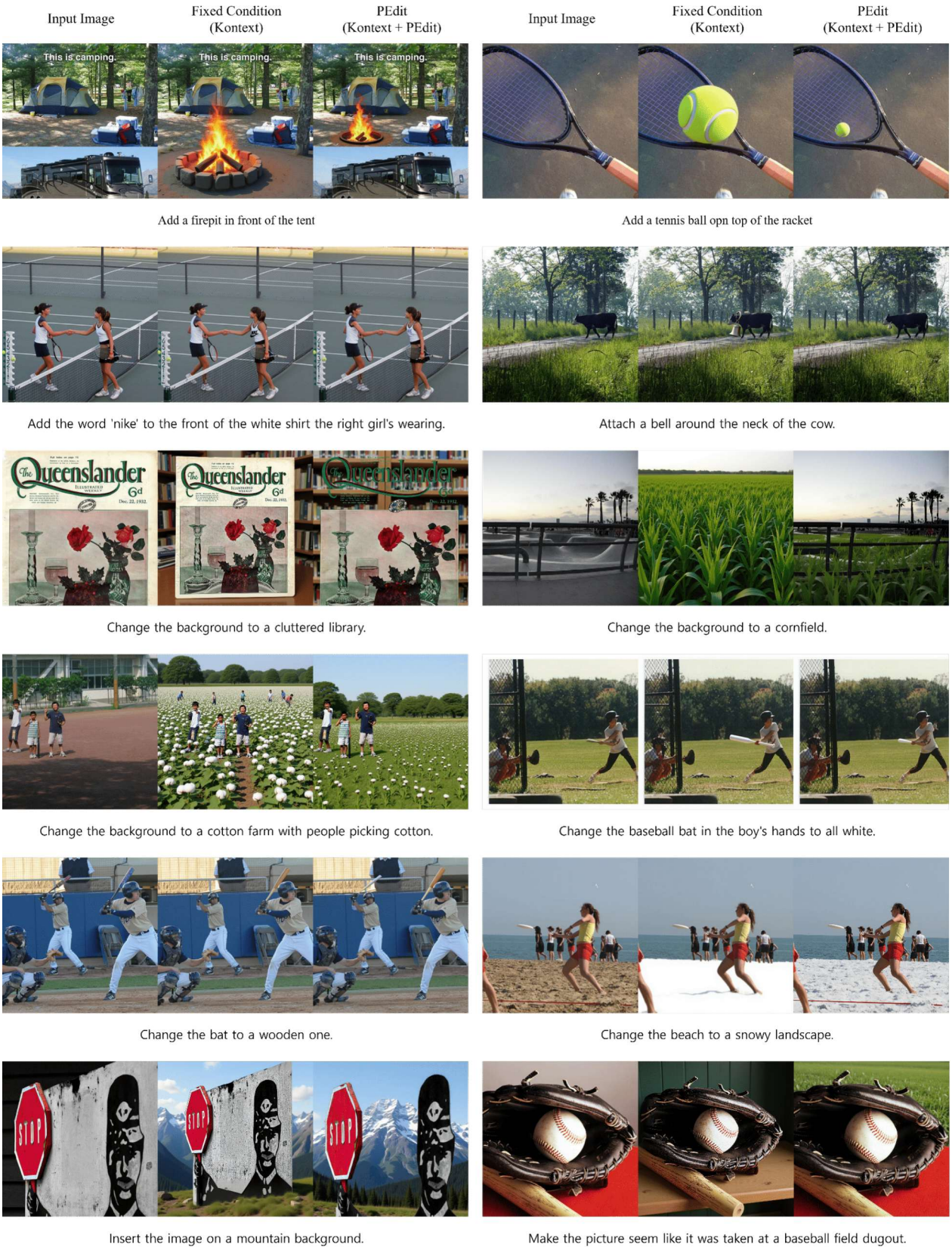


Figure F. Editing results of Kontext on the real image dataset of Emu-Edit, shown before and after applying PEdit.

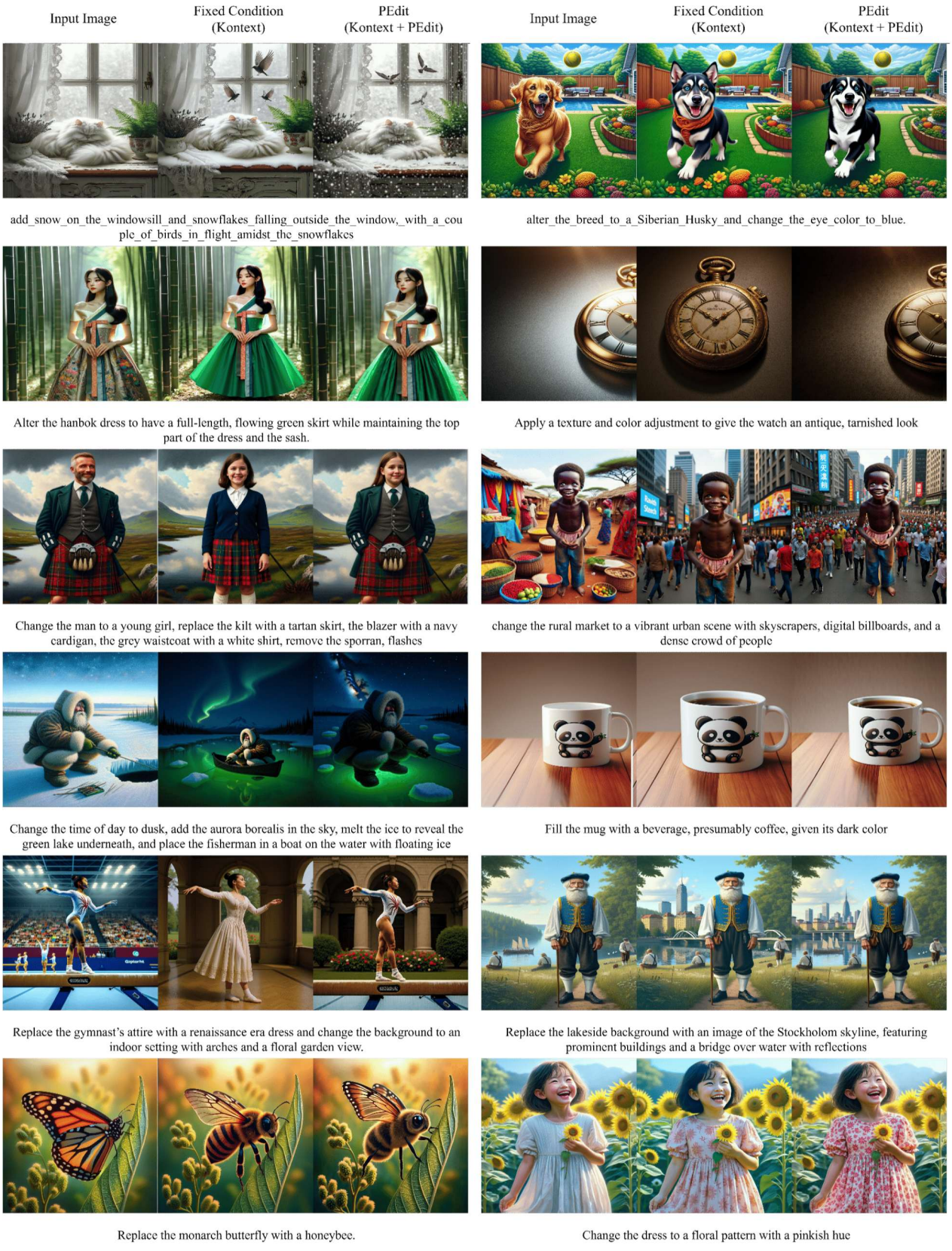


Figure G. Editing results of Kontext on the synthetic dataset of HQ-Edit, shown before and after applying PEdit.

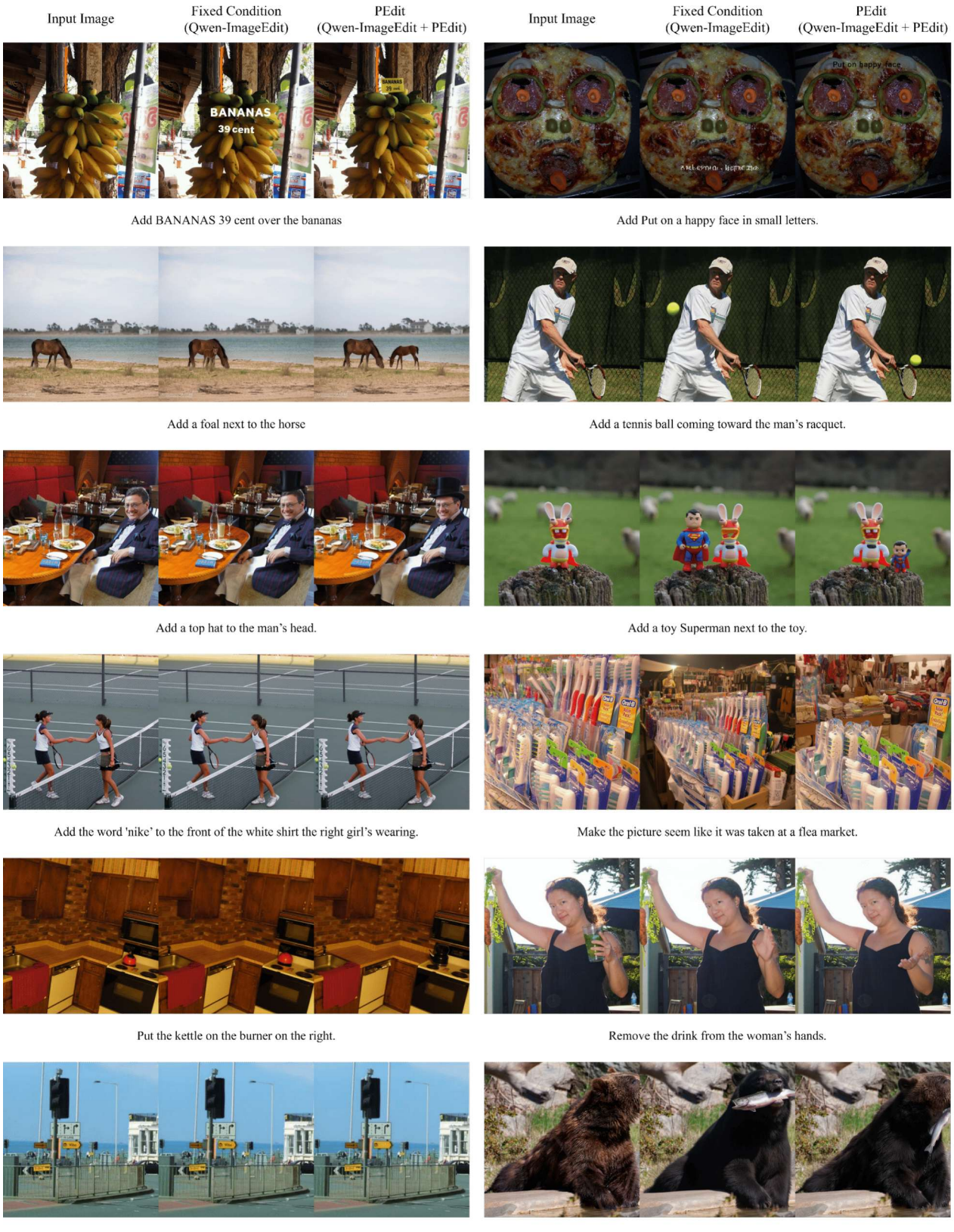


Figure H. Editing results of Qwen-ImageEdit on the real image Emu-Edit dataset, shown before and after applying PEdit.

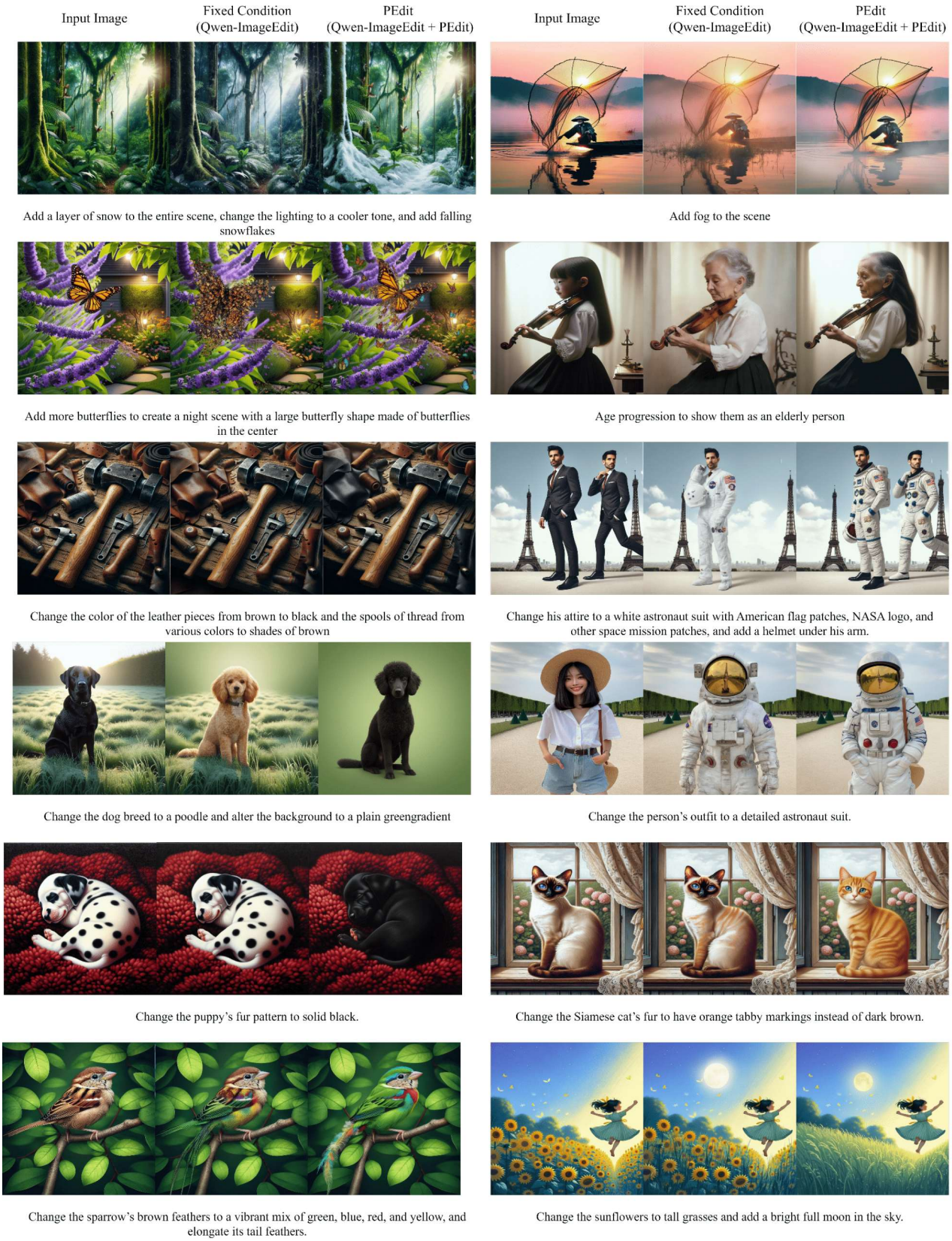


Figure I. Editing results of Qwen-ImageEdit on the synthetic dataset of HQ-Edit, shown before and after applying PEdit.