MotionEdit: Benchmarking and Learning Motion-Centric Image Editing

Supplementary Material

In this supplementary material, we present additional details on our method design in Section 1. We also present additional experimental setups, metric design, and human validation of metric in Section 2. Furthermore, we provide results of ablation experiments in Section 3.1 to verify the effectiveness of our MOTIONNFT method. Lastly, we visualize qualitative examples comparing our method to both open-source and closed-source commercial models in 3.2, highlighting the failure cases in these models and pointing towards future research direction.

1. Additional Method Details

1.1. Preliminaries

In T2I diffusion models, the forward noising process perturbs clean data x_0 from real distribution π_0 by adding a scheduled Gaussian noise $\epsilon \sim \mathcal{N}(0, \mathrm{I})$. The model then learns to reverse the noise and output clean images. The shift from Denoising Diffusion Probabilistic Models (DDPMs) to Flow Matching models (FMMs) is essentially a change in the prediction target of the model, from predicting the added noise itself (DDPM) to estimating a "velocity field" from the noise sample to the clean sample (FMM).

In mathematics formulation, FMMs define $z_t = \alpha_t x_0 + \sigma_t \epsilon$ to be a noisy interpolated latent at timestep t between the initial clean x_0 and the noise ϵ, where α_t and σ_t defines the scheduled noise at t. Then, for noisy sample z_t and textual context c, a FMM v_θ is trained to directly approximate the target constant velocity field $v = \frac{\mathrm{d}\alpha_t}{\mathrm{d}t} x_0 + \frac{\mathrm{d}\sigma_t}{\mathrm{d}t} \epsilon$ by minimizing the objective:

$$\mathcal{L}_{\text{FM}}(\theta) = \mathbb{E}_{t, x_0 \sim \pi_0, \epsilon \sim \mathcal{N}(0,\mathrm{I})} \left[\| v_\theta(\mathbf{x}_t, t, c) - v \|_2^2 \right].$$

This velocity prediction allows for efficient inference by solving the deterministic Ordinary Differential Equation (ODE), $dz_t = v_\theta(z_t, t, c)dt$ for the forward process.

1.2. Diffusion Negative-aware Finetuning (NFT)

DiffusionNFT [14] aims at finding not only the "positive velocity" $v^*(x_t, t, c) = v^+(x_t, t, c)$ that the model learns to predict, but also identifying the "negative velocity" $v^-(x_t, t, c)$ component that the model should steer away from. The training objective of DiffusionNFT is:

$$\begin{aligned} \mathcal{L}(\theta) = \mathbb{E}_{c,\, \pi^{old}(x_0|c),t}\; & r \left\| v_\theta^+(x_t, c, t) - v \right\|_2^2 \\ & + (1 - r) \left\| v_\theta^-(x_t, c, t) - v \right\|_2^2 \end{aligned} \tag{1}$$

Where v is the target velocity, v_θ^+ and v_θ^- are the implicit positive policy and implicit negative policy, defined as combinations of the old policy and current training policy:

$$\begin{aligned} v_\theta^+(x_t, c, t) &:= (1 - \beta)\, v^{old}(x_t, c, t) + \beta\, v_\theta(x_t, c, t), \\ v_\theta^-(x_t, c, t) &:= (1 + \beta)\, v^{old}(x_t, c, t) - \beta\, v_\theta(x_t, c, t). \end{aligned} \tag{2}$$

Naturally, we need an optimal reward r to accurately estimate the likelihood of the current action to fall into the "positive" subset of all samples. However, real-world reward models might differ in score distributions and scales. To this end, DiffusionNFT transforms raw rewards r^{raw} into the optimality reward:

$$r(x_0, c) := \frac{1}{2} + \frac{1}{2} \,\mathrm{clip}\left[\frac{r^{\mathrm{raw}}(x_0, c) - \mathbb{E}_{\pi^{old}(\cdot|c)} r^{\mathrm{raw}}(x_0, c)}{Z_c}, -1, 1 \right] \tag{3}$$

Where Z_c is a normalizing factor (e.g. standard deviation of global rewards).

1.3. MotionNFT: Motion-Aware Reward for NFT

We propose a optical flow-based **motion-centric reward scoring framework** for our **MotionNFT** method to compute how closely model-predicted motion matches the ground-truth motion. Our reward scoring process is illustrated as follows:

1.3.1. Motion Calculation

Optical Flow Calculation Given two images $\mathbf{I}_0$ and $\mathbf{I}_1$, an optical flow estimation model [11] $\mathcal{F}$ quantifies motion flow between them with $\mathbf{V} = \mathcal{F}(\mathbf{I}_{\text{orig}}, \mathbf{I}_{\text{edited}}) \in \mathbb{R}^{H \times W \times 2}$, where $\mathbf{V}$ is a vector field that represents the motion of each pixel with a 2D vector. In our case, given an input triplet $\mathbf{X} = (\mathbf{I}_{\text{orig}}, \mathbf{I}_{\text{edited}}, \mathbf{I}_{\text{gt}})$ containing triplets of the original image $\mathbf{I}_{\text{orig}}$, the model-edited image $\mathbf{I}_{\text{edited}}$, and the ground truth image $\mathbf{I}_{\text{gt}}$, we first calculate the optical flow between the input image and the model-edited image: $\mathbf{V}_{\text{pred}} = \mathcal{F}(\mathbf{I}_{\text{orig}}, \mathbf{I}_{\text{pred}})$. Then, we construct the motion reward $\mathbf{r}_m(\mathbf{X})$ to quantify the level of alignment between $\mathbf{V}_{\text{pred}}$ and the ground truth motion flow derived from the input and the ground-truth edited image $\mathbf{V}_{\text{gt}} = \mathcal{F}(\mathbf{I}_{\text{orig}}, \mathbf{I}_{\text{gt}})$ with three consistency terms: a *motion magnitude consistency* term, a *motion direction consistency* term, and an *movement regularization* term.

Flow normalization. Let $\mathbf{V}_{\text{pred}}(i, j) \in \mathbb{R}^2$ and $\mathbf{V}_{\text{gt}}(i, j) \in \mathbb{R}^2$ denote the optical flow vectors at pixel (i, j) for the model-edited and ground-truth edited images, respectively. For an image of height H and width W, we normalize the flows by the image diagonal to make the displacement magnitude comparable across resolutions:

$$\tilde{\mathbf{V}}_{\text{pred}}(i, j) = \frac{\mathbf{V}_{\text{pred}}(i, j)}{\sqrt{H^2 + W^2}}, \tilde{\mathbf{V}}_{\text{gt}}(i, j) = \frac{\mathbf{V}_{\text{gt}}(i, j)}{\sqrt{H^2 + W^2}}.$$

1.3.2. Reward Calculation

Motion Magnitude Consistency Term. We first measure how closely the predicted flow magnitudes match the ground truth using a robust ℓ_1 distance. Let $\mathbf{d}(i,j) = \tilde{\mathbf{V}}_{\text{pred}}(i,j) - \tilde{\mathbf{V}}_{\text{gt}}(i,j)$, magnitude deviation $\mathcal{D}_{\text{mag}}$ can be calculated as:

$$\mathcal{D}_{\text{mag}} = \frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W}(\|\mathbf{d}(i,j)\|_1 + \varepsilon)^q,$$

where $\varepsilon > 0$ is a small constant used for numerical stability and the exponent $q \in (0,1)$ enables a *robust* variant of the ℓ_1 distance that suppresses the influence of large outliers in the flow field while still preserving sensitivity to semantically meaningful deviations. Empirically, we set $q = 0.4$, which provides a stable trade-off between robustness and sensitivity for the motion-editing task.

Motion Direction Consistency Term. We additionally measure directional alignment between the two flow fields, while focusing on regions with non-trivial motion. Let:

$$m_{\text{gt}}(i,j) = \|\tilde{\mathbf{V}}_{\text{gt}}(i,j)\|_2,\ m_{\text{pred}}(i,j) = \|\tilde{\mathbf{V}}_{\text{pred}}(i,j)\|_2,$$

and define unit flow directions:

$$\hat{\mathbf{v}}_{\text{gt}}(i,j) = \frac{\tilde{\mathbf{V}}_{\text{gt}}(i,j)}{m_{\text{gt}}(i,j)+\varepsilon},\ \hat{\mathbf{v}}_{\text{pred}}(i,j) = \frac{\tilde{\mathbf{V}}_{\text{pred}}(i,j)}{\|\tilde{\mathbf{V}}_{\text{pred}}(i,j)\|_2+\varepsilon}.$$

We compute a cosine-based directional error per pixel:

$$\cos(i,j) = \hat{\mathbf{v}}_{\text{pred}}(i,j)^\top \hat{\mathbf{v}}_{\text{gt}}(i,j),\ e_{\text{dir}}(i,j) = \tfrac{1}{2}(1-\cos(i,j)),$$

and weight each pixel by the relative ground-truth motion magnitude:

$$w(i,j) = \frac{m_{\text{gt}}(i,j)}{\max_{u,v} m_{\text{gt}}(u,v)+\varepsilon}\cdot \mathbf{1}[m_{\text{gt}}(i,j) > \tau_m],$$

where τ_m is a small motion threshold and $\mathbf{1}[\cdot]$ is the indicator function. The directional misalignment $\mathcal{D}_{\text{dir}}$ can be calculated as:

$$\mathcal{D}_{\text{dir}} = \frac{\sum_{i,j} w(i,j)\, e_{\text{dir}}(i,j)}{\sum_{i,j} w(i,j)+\varepsilon}.$$

Movement Regularization Term. While $\mathcal{D}_{\text{mag}}$ and $\mathcal{D}_{\text{dir}}$ encourage the predicted flow to match the ground-truth motion, they do not by themselves prevent the model from collapsing to a nearly static edit. To discourage models from demonstrating this degeneration, we introduce a movement regularization term that compares the average motion magnitude of the predicted flow to that of the ground truth. We obtain the spatial means of m_{gt} and m_{pred}:

$$\bar{m}_{\text{gt}} = \frac{1}{HW}\sum_{i,j} m_{\text{gt}}(i,j),\ \bar{m}_{\text{pred}} = \frac{1}{HW}\sum_{i,j} m_{\text{pred}}(i,j),$$

and define the *anti-identity* hinge term to be:

$$M_{\text{move}} = \max\{0,\ \tau + \tfrac{1}{2}\,\bar{m}_{\text{gt}} - \bar{m}_{\text{pred}}\},$$

where $\tau > 0$ is a small margin. Intuitively, M_{move} penalizes trivial edits that keep the image nearly identical to the input.

Final: Motion-Centric Reward for training. Finally, we convert the optical flow-based alignment measure into a scalar reward for NFT training. We combine the 3 terms to obtain:

$$\mathcal{D}_{\text{comb}} = \alpha\,\mathcal{D}_{\text{mag}} + \beta\,\mathcal{D}_{\text{dir}} + \lambda_{\text{move}}\,M_{\text{move}},$$

where α, β, and λ_{move} are hyper-parameters that balance the scales between magnitude and directional alignments, as well as assigning a small proportion to discouraging under-motion. We normalize and clip the combined term:

$$\tilde{D} = \text{clip}\left(\frac{\mathcal{D}_{\text{comb}} - \mathcal{D}^*_{\text{min}}}{\mathcal{D}_{\text{max}} - \mathcal{D}^*_{\text{min}}},\ 0,\ 1\right),$$

where $\mathcal{D}^*_{\text{min}}$ is the lower bound of magnitude and directional terms calculated from a pair of duplicated inputs. We then construct the continuous optical flow-based reward:

$$r_{\text{cont}} = 1 - \tilde{L} \in [0,1],$$

so that higher reward corresponds to better alignment with the ground-truth motion edit. Finally, to approximate discrete human ratings of edited images following [5, 7], we quantize the reward to 6 equally spaced levels:

$$r_{\text{final}} = \frac{1}{5}\,\text{round}(5\,r_{\text{cont}}) \in \{0.0,\ 0.2,\ 0.4,\ 0.6,\ 0.8,\ 1.0\},$$

which is the final scalar reward signal for MotionNFT. During training, this raw reward score is further transformed to optimality rewards through group-wise normalization, and used to update the policy model v_θ by optimizing the DuffusionNFT objective in Equation 1.

2. Additional Evaluation Experiment Details

2.1. Hyperparameter Setting

2.1.1. Reward and Metric

MotionNFT Reward When calculating reward used for MOTIONNFT, we utilize three hyper-parameters to balance the three reward terms: $\mathcal{D}_{\text{comb}} = \alpha\,\mathcal{D}_{\text{mag}} + \beta\,\mathcal{D}_{\text{dir}} + \lambda_{\text{move}}\,M_{\text{move}}$ In our experiments, we set $\alpha = 0.7$, $\beta = 0.2$, and $\lambda_{\text{move}} = 0.1$. Not only does this balance the scales between magnitude and directional alignments, as well as assigning a small proportion to discouraging under-motion.

MAS Calculation When quantifying the MAS between model-edited images and ground truth targets, we punish degenerate cases where the predicted motion is nearly static compared to the ground-truth motion as a hard failure case and assign the minimum score MAS $= 0$. This happens when $\frac{\mathbb{E}[m_{\text{pred}}]}{\mathbb{E}[m_{\text{gt}}]} < \rho_{\min}$), where $\rho_{\min}$ is a parameter determining how harsh the punishment threshold would be. In our experiments, we set $\rho_{\min} = 0.01$.

2.1.2. Model Training

Following the training setup in Lin et al. [5], we train all models with learning rate set to $3e-4$. For FLUX.1 Kontext [Dev] [4] as base model, we report results for 300 steps. For Qwen-Image-Edit [10] as base model, we report results for 210 steps. Due to computational limits, we set batch size to 2. For NFT training, during sampling, we set sampling inference steps to 6, number of images per prompt to 8, and number of groups to 24; for training, we set KL loss' weight to 0.0001 and guidance strength to 1.0. For group filtering, we set the ban mean threshold to 0.9 and the standard deviation threshold to 0.05.

2.1.3. Model Inference

During inference for Qwen-Image-Edit [10] and the trained checkpoints, we set number of inference steps to 28, true cfg scale to 4.0, and guidance scale to 1.0. For inference of FLUX.1 Kontext [Dev] [4] and its trained checkpoints, we set the same number of inference steps. For inference of other open-sourced models, we follow the hyper-parameter setup in the official repositories. For UniWorld [5], we set number of inference steps to 25 and guidance scale to 3.5. For AnyEdit [12], we set guidance scale to 3, number of inference steps to 100, and original image guidance scale to 3. For UltraEdit [13], we set number of inference steps to 50, image guidance scale to 1.5, and guidance scale to 7.5. For Step-1X [6], we set number of inference steps to 28 and true cfg scale to 6.0. For all other models, we set guidance scale to 3.5 and number of inference steps to 28.

2.2. Human Evaluation for Generative Metrics

To evaluate the alignment between the MLLM-based generative metric and human judgment on motion editing quality, we conduct a human annotation study. Our annotators are a group of voluntary participants who are college-level or graduate-level students based in the United States. All annotators are proficient in English and have prior familiarity with AI research, ensuring that they understand the evaluation criteria and the purpose of the study. Prior to beginning, all annotators were informed that the anonymized results of their annotations may be used for research purposes only.

2.2.1. Annotation Interface and Instructions

We randomly sample 100 entries from MOTIONEDITBENCH for human evaluation, for which we further conduct random selection of outputs from 2 different models for comparison. To ensure consistency, all annotators completed the same set of comparison tasks. Each annotation instance consists of five visual components: (1) the *Input Image* to be edited, (2) the *Ground Truth Edited Image* demonstrating the ideal motion change, (3) a *Text Editing Instruction*, and (4–5) two model-generated edited outputs (labeled *Model 1* and *Model 2*). The annotators were asked to select which of the two model outputs better fulfill the requested motion edit, preserve the subject's identity, and maintain overall visual coherence. Annotators were reminded that the Ground Truth serves as a reference only, not something to be matched pixel-wise. They were encouraged to evaluate edits based on correctness of motion transformation and appearance preservation of the final image. If both outputs appear to be comparably good, annotators were instructed to select the one that is *slightly better*.

Annotation Instruction. Before beginning annotation, participants read the following notice and instructions:

> ***Warning:*** *The set of model-synthesized images displayed below might contain explicit, sensitive, or biased content.*
>
> *Thank you for being a human annotator for our study on the motion image editing task! By completing this form, you confirm voluntary participation in our research and agree to share your annotation data for research purposes only.*
>
> *For each example, you will see: the Input Image, the Ground Truth Edited Image, an Editing Instruction, and two model-generated outputs. Your task is to determine which model output better follows the editing instruction* ***while preserving the identity and appearance of the subject****. Consider whether the edit is applied correctly, whether the subject remains consistent with the input, and whether the final image appears coherent and natural. You may consider the Ground Truth Image to be a "reference answer" of the ideal edit. If both outputs are similar in quality, choose the one you feel is slightly better.*

2.2.2. Human Evaluation Results.

Since all annotatros complete the same set of comparison tasks, each pair of model comparison was labeled by three independent annotators. Inter-annotator agreement between all human annotators, as measured by Fleiss' κ, is 0.607, indicating **good agreement** among human raters. The aggregated agreement between human annotators and decisions made by the overall generative metric (averaged over Fidelity, Preservation, and Coherence) achieves a **Fleiss' κ score of 0.574, similarly demonstrating substantial alignment between human judgment and our metric**. These results support the use of the MLLM-based generative evaluation metric as a practical and human-consistent measure of motion editing quality.

3. Additional Evaluation Results

3.1. Ablation Studies

Balancing MLLM and Optical Flow-Based Rewards We investigate the optimal balancing strategy between our pro-

posed optical flow-based motion alignment reward (r_{motion}) and the MLLM-based semantic reward (r_{mllm}) introduced in Uniworld-v2. Specifically, we adopt multi-objective reward NFT training with different weights for each reward. Table 1 summarizes the editing performance on our MOTIONEDIT-BENCH across varying balancing weights. We observe that relying solely on the motion reward (1.0 * *Motion*) leads to a performance degradation, indicating that geometric motion cues alone are insufficient for maintaining semantic fidelity. Conversely, while the pure MLLM reward (1.0 * *MLLM*) provides a strong baseline, it is consistently outperformed by the combined approach. The results demonstrate that the two objectives are complementary. The balanced configuration ($\lambda = 0.5$) yields the highest performance across all metrics for both FLUX.1 Kontext [Dev] [4] and Qwen-Image-Edit [10] backbones (achieving 4.25 and 4.72 Overall scores, respectively). This suggests that the optical flow reward effectively regularizes the MLLM guidance, improving motion alignment without compromising semantic coherence.

Model	MotionEdit-Bench			
	Ovl. ↑	Fid. ↑	Pre. ↑	Coh. ↑
FLUX.1 Kontext	3.84	3.89	3.79	3.83
+ *1.0 * Motion*	3.60	3.62	3.60	3.59
+ *0.3 * MLLM + 0.7 * Motion*	4.22	4.29	4.15	4.23
+ *0.7 * MLLM + 0.3 * Motion*	4.16	4.23	4.08	4.16
+ *1.0 * MLLM*	4.20	4.28	4.11	4.21
+ 0.5 * MLLM + 0.5 * Motion	**4.25**	**4.33**	**4.16**	**4.25**
Qwen-Image-Edit	4.65	4.70	4.59	4.66
+ *1.0 * Motion*	4.60	4.65	4.55	4.61
+ *0.3 * MLLM + 0.7 * Motion*	4.72	**4.81**	4.61	4.74
+ *0.7 * MLLM + 0.3 * Motion*	4.71	4.78	4.62	4.73
+ *1.0 * MLLM*	4.70	4.80	4.57	4.73
+ 0.5 * MLLM + 0.5 * Motion	**4.72**	4.79	**4.63**	**4.74**

Table 1. Ablation experiments on different weights for balancing the MLLM-based reward proposed by [5] and our optical flow-based motion alignment reward. Results show that combining both rewards on a 0.5:0.5 scale achieves best performance, outperforming MLLM-only reward training.

MLLM-only Reward vs. MotionNFT Figures 1 and 2 visualize the evolution of the Motion Alignment Score (MAS) during training for the MLLM-only reward in Lin et al. [5] and our MOTIONNFT reward. As explained in previous sections, MAS utilizes optical flow to quantify magnitude and directional alignment level between model-edited motion and ground truth motion achieved in the target image. We observe that relying solely on the MLLM-based semantic reward results in suboptimal motion alignment; for Qwen-Image-Edit (Fig. 2), the MAS even degrades significantly during the mid-training phase. In contrast, MotionNFT demonstrates robust and consistent improvement in MAS across both backbone models. By incorporating explicit motion guidance, our method prevents the model from overfitting to semantic cues at the expense of geometric accuracy, ultimately achieving a significantly higher final MAS.

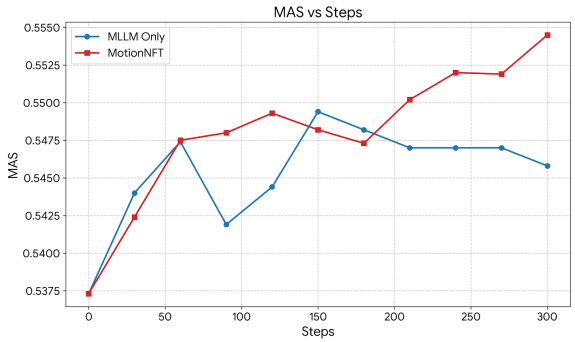

Figure 1. MAS vs. Training Steps on FLUX.1 Kontext [Dev] [4]. MAS quantifies the fidelity of the generated motion by calculating the optical flow alignment (considering both magnitude and direction) between the model's edit and the ground truth target edit. While the MLLM-only baseline (blue) begins to regress after approximately 150 steps, MotionNFT (red) demonstrates steady improvement throughout training, ultimately achieving superior motion grounding by leveraging explicit motion guidance.

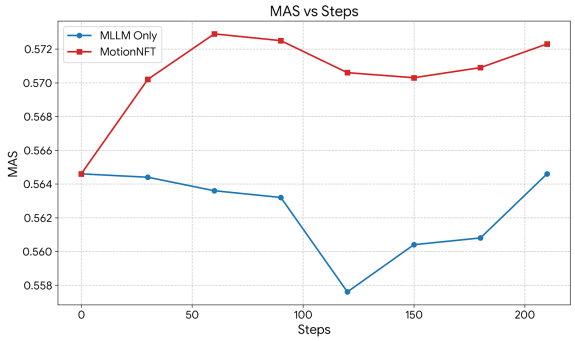

Figure 2. MAS vs. Training Steps on Qwen-Image-Edit [10]. Results on the other base model again shows that relying solely on semantic MLLM rewards leads to training regression in motion alignment. Again, MotionNFT maintains a stable upward trajectory, preventing overfitting to semantic cues and achieving a higher final MAS.

Qualitative Examples Figures 3 and 4 compare MotionNFT against the base models (Qwen-Image-Edit [10] and FLUX.1 Kontext [Dev] [4]) and their MLLM reward [5]-guided counterparts. A recurring failure mode in the baselines is lack of "motion awareness", where the model fails to interpret and execute the desired motion subject, direction, and magnitude from the editing prompts. For instance, in Figure 3 (Row 2), the Qwen baseline and the UniWorld-V2 baseline fails to correctly move the subject's right hand to operate the joystick, but rather placing both hands on it. In Row 6, both baselines mistakenly flip the caterpillar's body direction when moving it towards the center of the flower. In contrast, MotionNFT successfully executes both edits, matching the ground truth desired motion.

Additionally, we observe that another failure mode in baseline methods is the preservation of setting and subject

identity. In Figure 3 (Row 5), both baselines completely remove the milk jar despite it being a main subject in the image. In the last row, both baselines remove the photo frame surrounding the woman that was in the original image, failing to preserve setting consistency. Similarly, in Figure 4 row 2, we observe that using [5]'s MLLM-only reward on FLUX.1 Kontext changes the penguin's beak in to a black color, failing to preserve its appearance while also not correctly performing the motion edit. MotionNFT, on the other hand, achieves good preservation of subject's appearance and setting consistency.

3.2. Model Comparison

3.2.1. Comparison with Open-Source Models

We compare MotionNFT against leading open-source editing models, including UniWorld-V1 [5], BAGEL [2], and FLUX.1 Kontext [Dev] [4]. Visual comparisons in Figure 5 reveal that these baselines frequently struggle with precise motion controllability:

- **Editing Inertia:** Existing models may fail to execute significant geometric transformations, defaulting to the original pose. For instance, in the "car cliff" scenario (Row 6), UniWorld-V1 fails to displace the vehicle, leaving it on the ledge with a flipped direction, while BAGEL and FLUX.1 lift the car but fail to capture the "downward angled" physics of the fall. Similarly, in the "lion" example (Row 2), all baselines fail to fully lower the head to the requested "looking downwards" pose, whereas MotionNFT achieves accurate alignment with the ground truth.
- **Motion Misalignment:** Existing models may fail to interpret and execute the subject part and direction of the motion change. For instance, in the gorilla example (Row 3), FLUX.1 Kontext fails to put the right hand into a fist. In the robot example (row 5), all baseline models fail to move the robot's left arm but move the right one instead. MotionNFT, on the other hand, performs the correct motion change on the correct subject part.
- **Structural Distortion:** When baselines do attempt large edits, they often introduce anatomical or semantic artifacts. In the "gorilla" example (Row 3), FLUX.1 Kontext distort the hand structure when attempting the "fist" gesture. In the jug drinking example (Row 4), the baselines leave residual artifacts that distorts the jug, while our method cleanly executes the edit without artifacts.

3.2.2. Comparison with Closed-source Models

We conduct selective case studies that compares MotionNFT with Qwen-Image-Edit as base model against leading closed-source commercial models, including Nano-Banana [3], GPT-Image-1 [8], Seedream [9], and Hunyuan Image [1]. As visualized in Figure 6, these models still exhibit distinct failure modes for motion editing:

1. **Semantic Hallucination and Structural Distortion:** When complex pose changes are required, baselines often introduce artifacts or unwanted semantic changes. In the "apple" example (Row 2), Nano-Banana introduces artifact by creating an additional "feet" that steps on the apple. MotionNFT avoids these structural collapses, successfully executing the editing instructions with high anatomical fidelity.
2. **Motion Misalignment:** Even strong closed-source commercial models suffer from correctly identifying the subject part, direction, and magnitude of the motion change. For instance, in the anime girl example (row 1), Nano-banana demonstrates editing inertia where the motion of the subject remains the same, while GPT-Image 1, Seedream, and Hunyuan Image fail to execute the correct edit on the girl's arm.

3.2.3. Failure Analysis and Limitations

While MotionNFT demonstrates robust performance across a wide range of editing cases, we observe specific scenarios where it, alongside leading commercial models, encounters difficulties. Figure 7 illustrates these common failure modes that highlights persisting challenges:

- **Multi-Subject Interactions:** Challenging editing settings with multiple involving and non-involving subjects in images pose a major challenge for existing models. For instance, in the orca example (row 1), all models fail to position the orca in front of the polar bear while executing the motion change to make it submerge in water. Similarly, in the crew member example (row 3), changing only the direction and the motion of one subject among a couple is difficult for existing models.
- **Identity Preservation** Existing models still struggle to preserve subjects and their identities in complex scenes. For instance, in the chicken example (row 2), 3 models fail to preserve the chicken's appearance. In the tent example (last row), models fail to preserve additional subjects in the scene not involved in the motion change.

These cases suggest that future work incorporating stronger physics-based priors or motion guidance could further resolve the remaining challenges.

3.3. Speed and Inference Cost

MotionNFT is designed to be lightweight and computationally efficient. A key advantage of our method is that it can be seamlessly integrated with base models such as FLUX.1 Kontext Dev and Qwen-Image-Edit with no additional inference-time cost. All experiments were conducted on a single NVIDIA GPU. Using 28 sampling steps for a single entry, inference requires approximately 48 seconds with the FLUX.1 backbone and 98 seconds with Qwen-Image-Edit. This confirms that MotionNFT enhances generation capabilities without compromising the speed or hardware requirements of the original models.

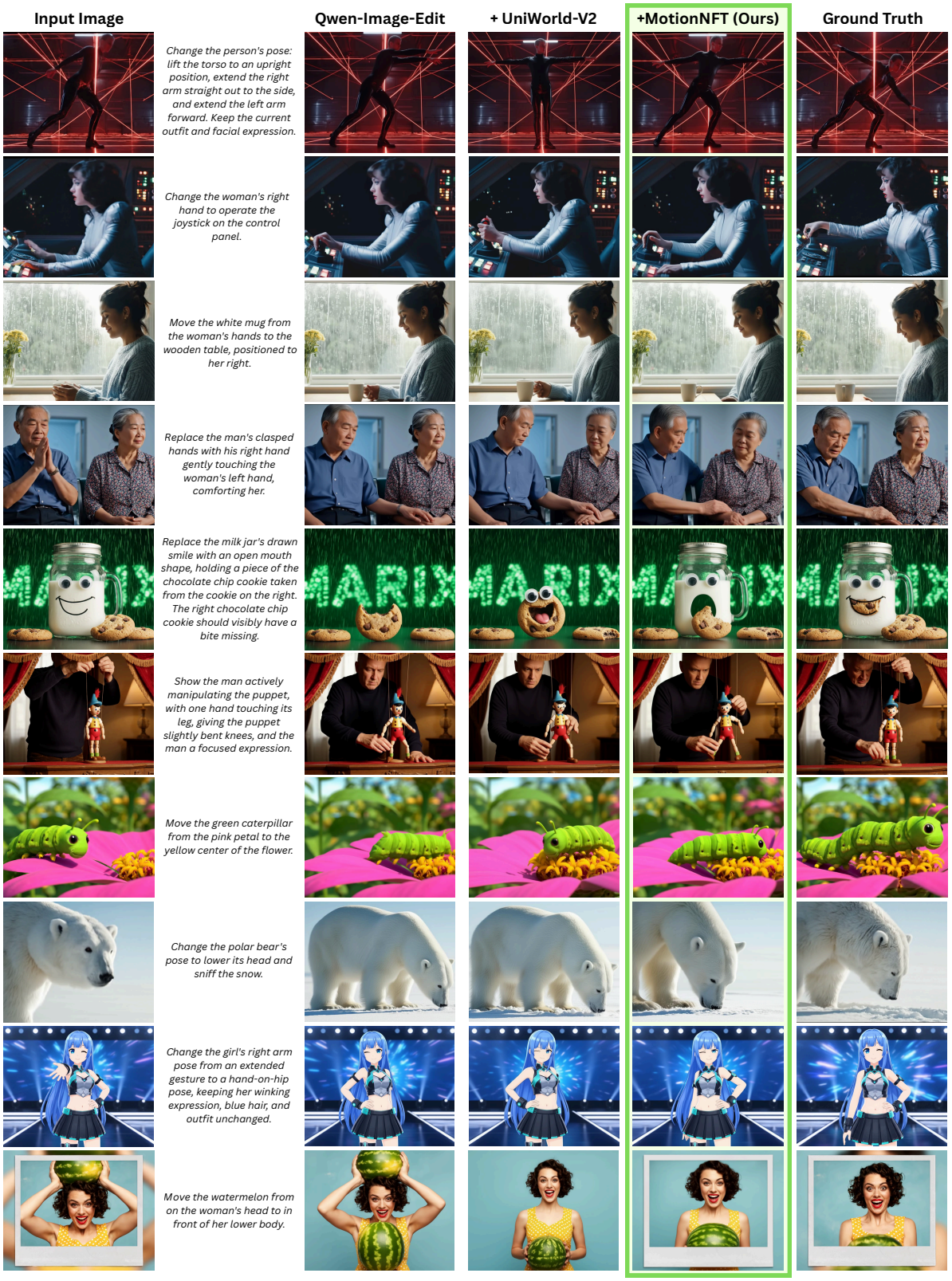

Figure 3. Qualitative comparison of our method to Qwen-Image-Edit [10] and the MLLM-only reward training in Lin et al. [5]. The base model frequently fails to demonstrate correct motion awareness for the edit (e.g. fail to move the subject's arms in the first row, and failing to displace the watermelon in the last row). While the MLLM-only approach improves semantic adherence, it often lacks geometric precision (e.g., caterpillar's orientation in row 7). MotionNFT leverages optical flow to bridge this gap, achieving precise motion alignment and high fidelity to the editing instructions.

Figure 4. Qualitative comparison of our method to FLUX.1 Kontext [Dev] [4] and the MLLM-only reward training in Lin et al. [5]. The base model often exhibits editing inertia, failing to execute structural changes such as removing the handshake (row 6) or changing the subjects' directions (row 3). MLLM-only baseline also frequently hallucinates incorrect poses (e.g., the distorted limb placement in row 5) or fails to preserve subject identity (row 2). MotionNFT is able to interpret and execute complex motion edit instructions while preserving subject appearance and visual setting.

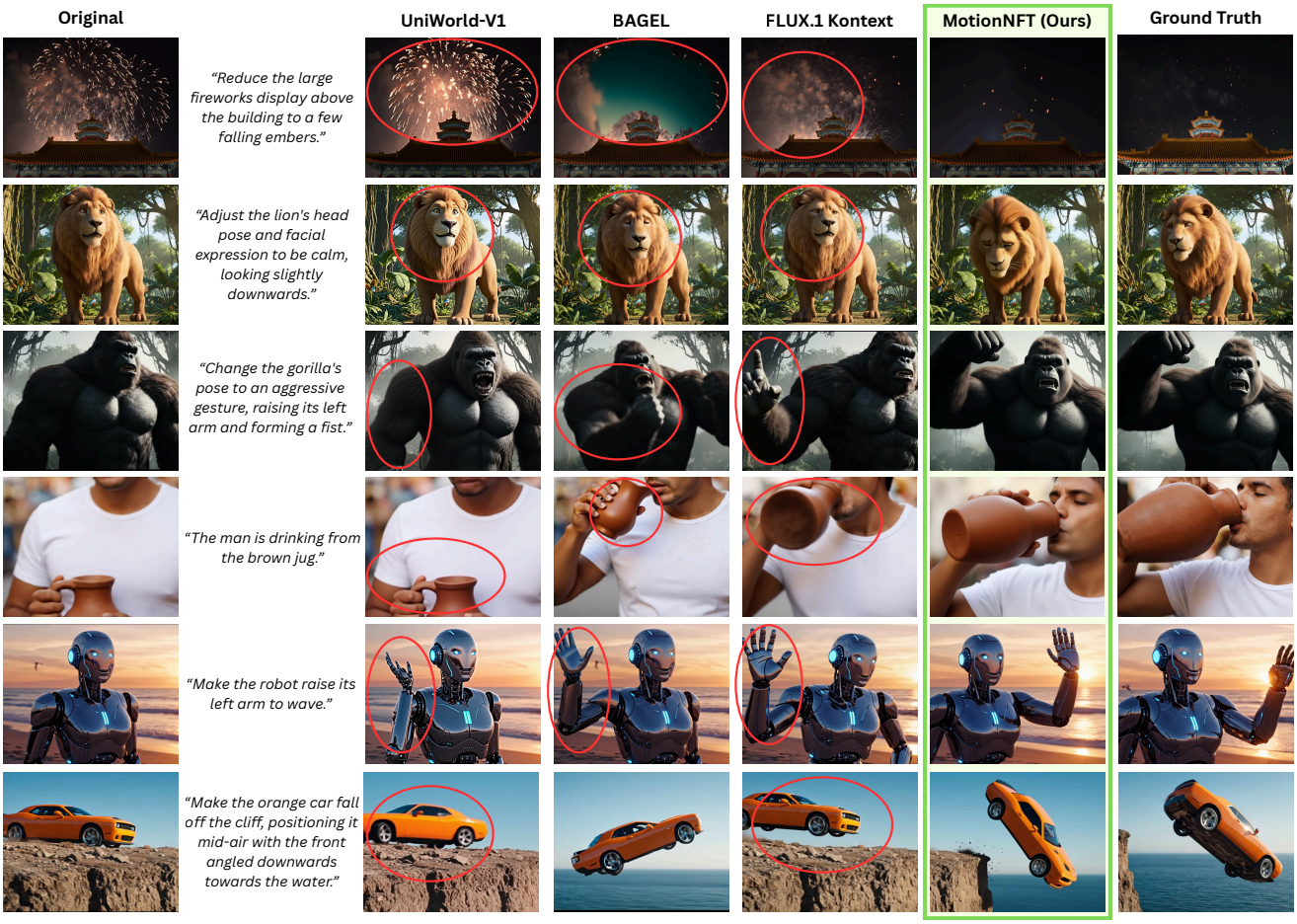

Figure 5. We compare MotionNFT against state-of-the-art baselines: UniWorld-V1 [5], BAGEL [2], and FLUX.1 Kontext [Dev] [4]. Red circles highlight failure regions. Baseline models exhibit different failure modes like editing inertia (e.g., failing to change the lion's pose in row 2), or motion misalignment (e.g., raising the robot's right arm instead of left arm in row 5). While baselines often struggle to execute challenging motion edits, MotionNFT achieves superior geometric grounding, accurately following semantic instructions and maintaining high motion fidelity to the ground truth.

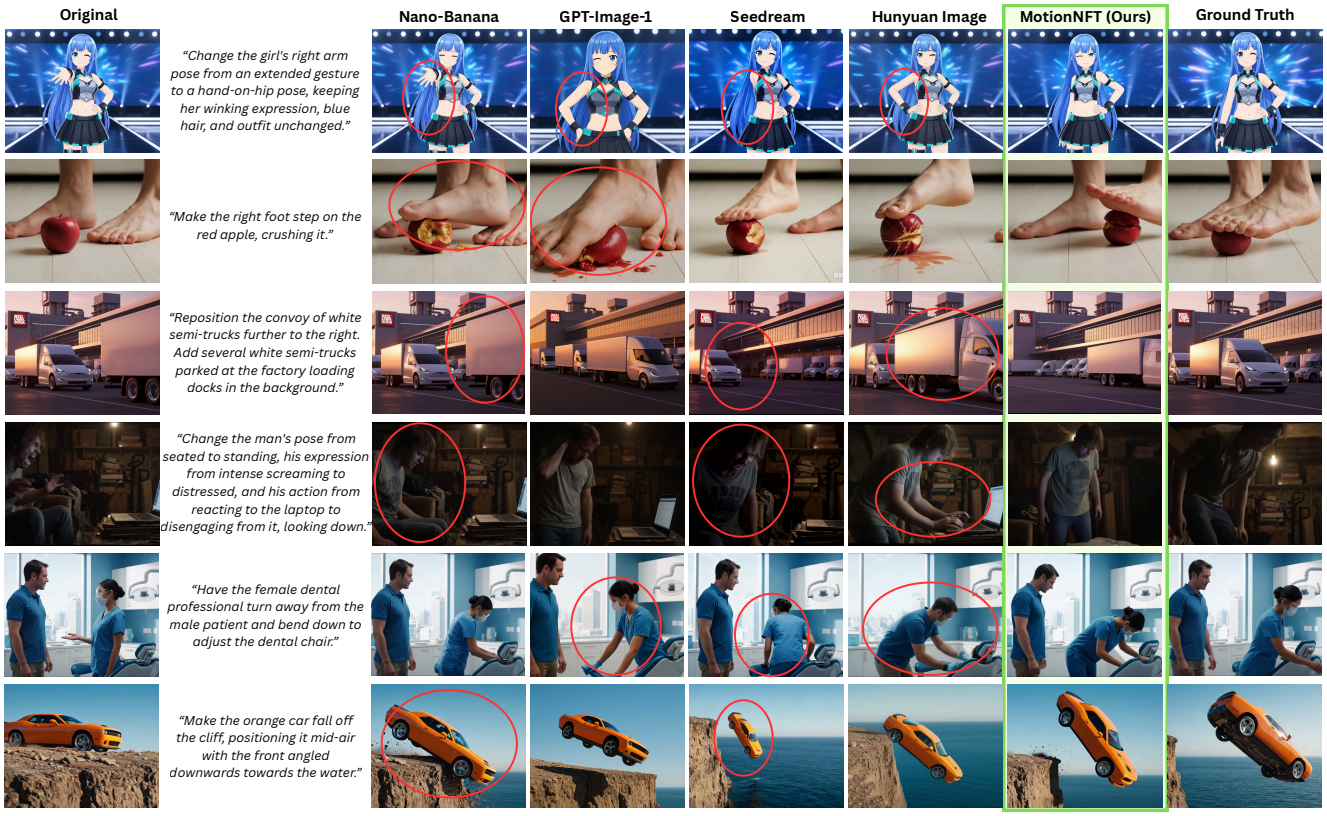

Figure 6. We conduct selective case studies of MotionNFT against leading closed-source commercial baselines: Nano-Banana [3], GPT-Image-1 [8], Seedream [9], and Hunyuan Image [1]. Red circles highlight failure regions where baselines exhibit spatial inertia (e.g., failing to displace the car in the bottom row) or structural hallucination (e.g., generating an artifact "foot" in the second row). While commercial models generally maintain high visual quality, they frequently struggle to ground complex motion changes or maintain visual consistency. MotionNFT accurately follows these dynamic instructions, ensuring geometric alignment with the ground truth.

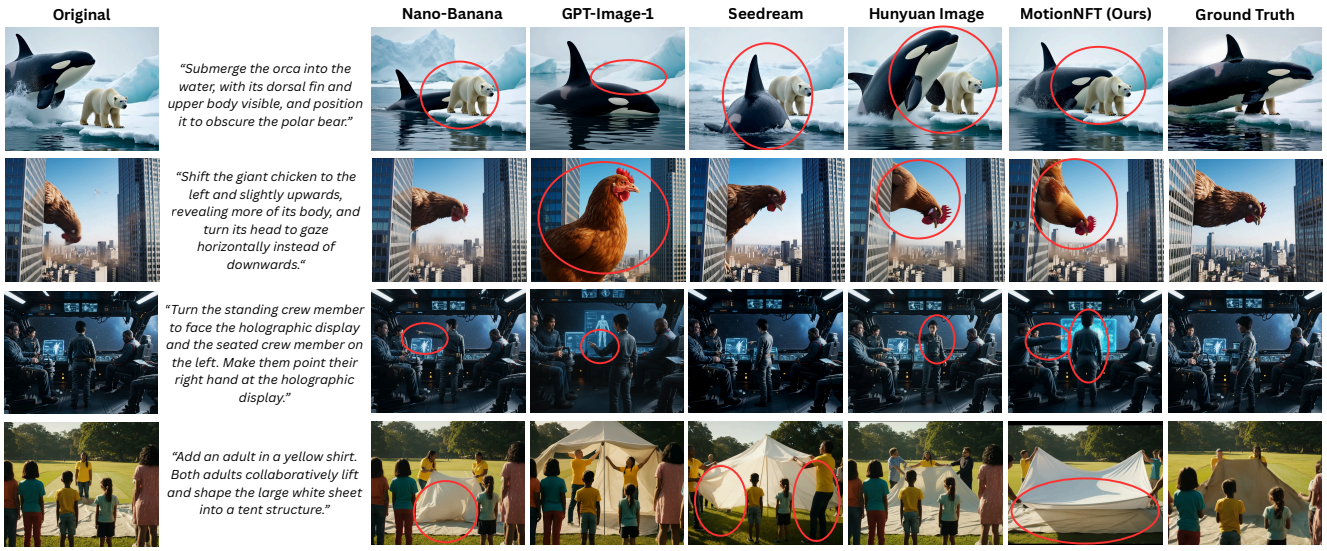

Figure 7. Additional failure cases of our model and closed-source commercial models. We observe that instructions involving multiple involving and non-involving subjects (e.g. the orca example in row 1, which requires complex 3D spatial edit) remain challenging for all evaluated methods. Current models, including ours and commercial baselines, struggle to correctly generate accurate and targeted motions on the correct subject part with the correct direction and magnitude in challenging scenarios.

References

[1] Siyu Cao, Hangting Chen, Peng Chen, Yiji Cheng, Yutao Cui, Xinchi Deng, Ying Dong, Kipper Gong, Tianpeng Gu, Xiusen Gu, et al. Hunyuanimage 3.0 technical report. *arXiv preprint arXiv:2509.23951*, 2025. 5, 9

[2] Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Guang Shi, and Haoqi Fan. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025. 5, 8

[3] Google. Gemini image generation api, 2025. https://ai.google.dev/gemini-api/docs/image-generation/. 5, 9

[4] Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Müller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, and Luke Smith. Flux.1 kontext: Flow matching for in-context image generation and editing in latent space, 2025. 3, 4, 5, 7, 8

[5] Bin Lin, Zongjian Li, Xinhua Cheng, Yuwei Niu, Yang Ye, Xianyi He, Shenghai Yuan, Wangbo Yu, Shaodong Wang, Yunyang Ge, et al. Uniworld: High-resolution semantic encoders for unified visual understanding and generation. *arXiv preprint arXiv:2506.03147*, 2025. 2, 3, 4, 5, 6, 7, 8

[6] Shiyu Liu, Yucheng Han, Peng Xing, Fukun Yin, Rui Wang, Wei Cheng, Jiaqi Liao, Yingming Wang, Honghao Fu, Chunrui Han, Guopeng Li, Yuang Peng, Quan Sun, Jingwei Wu, Yan Cai, Zheng Ge, Ranchen Ming, Lei Xia, Xianfang Zeng, Yibo Zhu, Binxing Jiao, Xiangyu Zhang, Gang Yu, and Daxin Jiang. Step1x-edit: A practical framework for general image editing. *arXiv preprint arXiv:2504.17761*, 2025. 3

[7] Xin Luo, Jiahao Wang, Chenyuan Wu, Shitao Xiao, Xiyan Jiang, Defu Lian, Jiajun Zhang, Dong Liu, et al. Editscore: Unlocking online rl for image editing via high-fidelity reward modeling. *arXiv preprint arXiv:2509.23909*, 2025. 2

[8] OpenAI. Image generation api, 2025. https://openai.com/index/image-generation-api/. 5, 9

[9] Team Seedream, Yunpeng Chen, Yu Gao, Lixue Gong, Meng Guo, Qiushan Guo, Zhiyao Guo, Xiaoxia Hou, Weilin Huang, Yixuan Huang, et al. Seedream 4.0: Toward next-generation multimodal image generation. *arXiv preprint arXiv:2509.20427*, 2025. 5, 9

[10] Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, Yuxiang Chen, Zecheng Tang, Zekai Zhang, Zhengyi Wang, An Yang, Bowen Yu, Chen Cheng, Dayiheng Liu, Deqing Li, Hang Zhang, Hao Meng, Hu Wei, Jingyuan Ni, Kai Chen, Kuan Cao, Liang Peng, Lin Qu, Minggang Wu, Peng Wang, Shuting Yu, Tingkun Wen, Wensen Feng, Xiaoxiao Xu, Yi Wang, Yichang Zhang, Yongqiang Zhu, Yujia Wu, Yuxuan Cai, and Zenan Liu. Qwen-image technical report, 2025. 3, 4, 6

[11] Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, Fisher Yu, Dacheng Tao, and Andreas Geiger. Unifying flow, stereo and depth estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 1

[12] Qifan Yu, Wei Chow, Zhongqi Yue, Kaihang Pan, Yang Wu, Xiaoyang Wan, Juncheng Li, Siliang Tang, Hanwang Zhang, and Yueting Zhuang. Anyedit: Mastering unified high-quality image editing for any idea. *arXiv preprint arXiv:2411.15738*, 2024. 3

[13] Haozhe Zhao, Xiaojian Ma, Liang Chen, Shuzheng Si, Rujie Wu, Kaikai An, Peiyu Yu, Minjia Zhang, Qing Li, and Baobao Chang. Ultraedit: Instruction-based fine-grained image editing at scale, 2024. 3

[14] Kaiwen Zheng, Huayu Chen, Haotian Ye, Haoxiang Wang, Qinsheng Zhang, Kai Jiang, Hang Su, Stefano Ermon, Jun Zhu, and Ming-Yu Liu. Diffusionnft: Online diffusion reinforcement with forward process. *arXiv preprint arXiv:2509.16117*, 2025. 1