MotionEdit: Benchmarking and Learning Motion-Centric Image Editing

Yixin Wan[1,2*], Lei Ke[1], Wenhao Yu[1], Kai-Wei Chang[2], Dong Yu[1]
[1]Tencent AI, Seattle [2]University of California, Los Angeles
Project Page: https://motion-edit.github.io

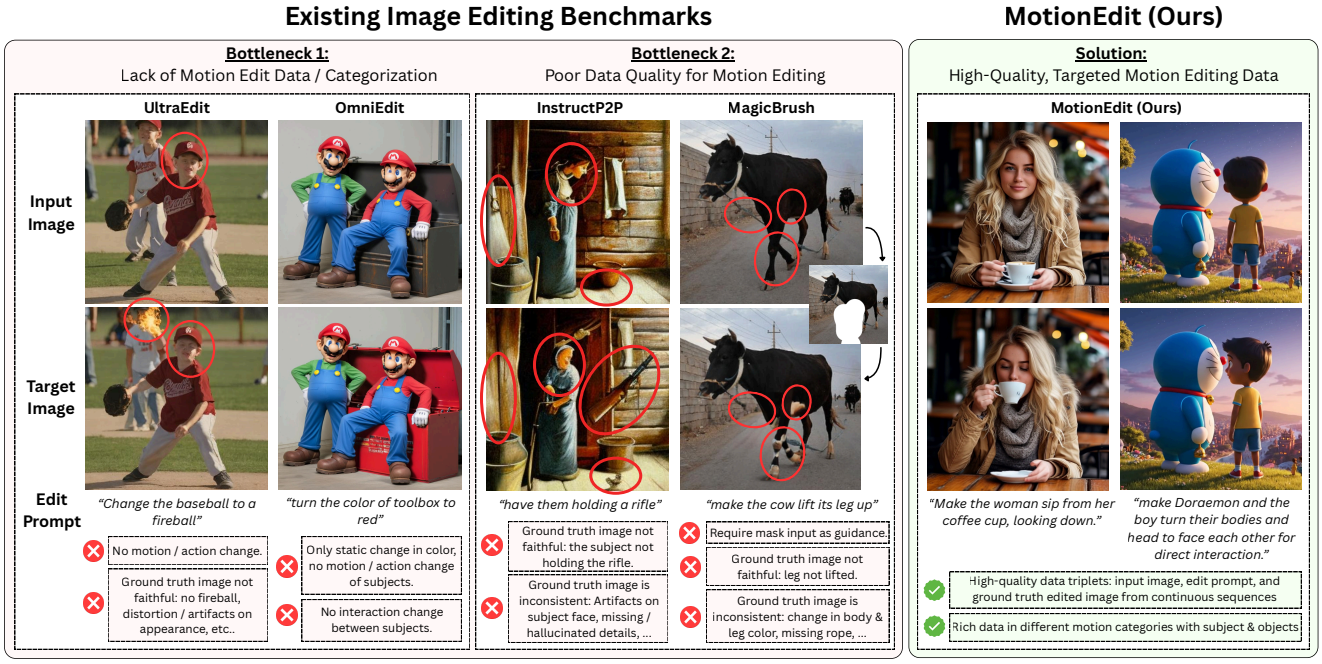

Figure 1. Comparison of existing image editing benchmarks with **MOTIONEDIT**. Prior datasets lack reliable motion-edit supervision—either focusing only on appearance edits or offering low-quality, inconsistent action changes with artifacts. **MOTIONEDIT** fills this gap by providing high-quality, instruction-following motion edits with paired input–target image data, enabling accurate evaluation and training of motion-aware image editing models.

Abstract

*We introduce **MotionEdit**, a novel dataset for motion-centric image editing—the task of modifying subject actions and interactions while preserving identity, structure, and physical plausibility. Unlike existing image editing datasets that focus on static appearance changes or contain only sparse, low-quality motion edits, MotionEdit provides high-fidelity image pairs depicting realistic motion transformations extracted and verified from continuous videos. This new task is not only scientifically challenging but also practically significant, powering downstream applications such as frame-controlled video synthesis and animation.*

*To evaluate model performance on the novel task, we introduce **MotionEdit-Bench**, a benchmark that challenges models on motion-centric edits and measures model performance with generative, discriminative, and preference-based metrics. Benchmark results reveal that motion editing remains highly challenging for existing state-of-the-art diffusion-based editing models. To address this gap, we propose **MotionNFT** (Motion-guided Negative-aware Fine-Tuning), a post-training framework that computes motion alignment rewards based on how well the motion flow between input and model-edited images matches the ground-truth motion, guiding models toward accurate motion transformations. Extensive experiments on FLUX.1 Kontext and Qwen-Image-Edit show that MotionNFT consistently improves editing quality and motion fidelity of both base models on the motion editing task without sacrificing general editing ability, demonstrating its effectiveness.*

*Work done during internship at Tencent AI Lab in Seattle, contact email: elaine1wan@cs.ucla.edu

1. Introduction

Instruction-guided image editing models have made remarkable progress recently [6, 12, 13, 20, 31], capable of transforming images based on natural language commands. While recent image editing models excel at performing appearance-only static edits that simply adjust color, texture, or object presence, they oftentimes fall short in accurately, faithfully, and naturally editing the motion, posture, or interaction between subjects in images. In this work, we aim at addressing this limitation in existing models through systematically formulating and studying motion editing as an independent and important image editing task.

We formally define the new task of **motion image editing**—editing that modifies the action, pose, or interaction of subjects and objects in an image according to a textual instruction, while preserving visual consistency in characters and scene. Motion editing aims at changing *how* subjects move, act, or interact, which is essential for applications such as frame-controlled video generation and character animation. However, existing image editing datasets and benchmarks suffer from two major bottlenecks in approaching the motion image editing task: First, they primarily focus on static editing tasks like appearance modification or replacement (e.g. OmniEdit [30] and UltraEdit [37] examples in Figure 1), neglecting the important aspect of motion editing in their data at all. Second, datasets that do include motion edits offer only a small amount of low-quality data, often with unfaithful or incoherent edit ground-truth that fail to execute the intended motion (e.g. InstructP2P [2] and MagicBrush [36] examples in Figure 1).

To bridge this research gap, we curate **MOTIONEDIT**, a high-quality dataset and benchmark specifically targeting motion editing, consisting of paired input–target image examples extracted and validated from continuous high-resolution video frames to ensure accurate, natural, and coherent motion changes. As shown in Figure 1, MOTIONEDIT captures realistic action and interaction changes that preserve identity, background, and style, in contrast to prior datasets where edit data is either static, unfaithful, or visually inconsistent. Moreover, our data is sourced from a large set diverse video sequences, ensuring the assessment of diverse sub-categories of motion image editing, such as posture, orientation, and interaction changes in Figure 4. Beyond constructing high quality editing data, we also devise evaluation metrics to evaluate motion edit performances of models. For discriminative evaluation, we by comparing the optical flow [9, 25, 27, 32, 33]—which captures the magnitude and direction of motion change—between the input and model-edited images against the input–ground truth flow. For generative evaluation, we adopt Multimodal Large Language Model (MLLM)-based metrics to assess the fidelity, preservation, coherence, and overall quality of edited images. Additionally, we report pairwise win rates through head-to-head comparisons between overall edit quality of different models to reflect preference performance. Both quantitative and qualitative results across state-of-the-art image editing models on MOTIONEDIT-BENCH show that **motion image editing remains a challenging task for the majority of open-source image editing models.**

To improve existing image editing models on the motion editing task, we further propose Motion-guided Negative-aware FineTuning (**MOTIONNFT**), a post-training framework for motion editing that extends DiffusionNFT [38] to incorporate motion-aware reward signals. MotionNFT leverages the motion alignment measurement between input-edit and input-ground truth optical flows to construct a reward scoring framework, providing targeted guidance on motion direction and magnitude in training. As illustrated in Figure 2, MotionNFT enables models to perform accurate, geometrically consistent motion edits. Quantitative results in Table 1 further shows substantial improvement across all metrics over prior approaches. For instance, MOTIONNFT achieves over 10% improvement in overall quality and over 12% on pairwise win rates when applied on FLUX.1 Kontext [12]).

The key contributions of our paper are three-fold:

- We systematically define and study the novel task of **motion image editing**.
- We construct **MOTIONEDIT**, a high quality dataset and benchmark for motion image editing, containing diverse and accurate edit data sourced from video frames.
- We propose **MOTIONNFT**, a post-training framework that integrates optical flow–based rewards into DiffusionNFT to guide motion edit improvements.

We have publicly released our code base[1], MotionEdit-Bench[2], as well as MotionEdit-Train[3].

2. Related Works

Image Editing. Recent advances in text-to-image (T2I) diffusion models have greatly improved text-guided image editing [2, 12, 18, 20, 31, 36]. While current models handle static appearance edits well (e.g., color changes or object replacement), they struggle with motion-related edits that require modifying actions or interactions (e.g., "make the man drink from the cup"). This gap largely stems from limitations in existing editing datasets. First, most benchmarks focus on static transformations—local texture changes, object replacement, or style transfer [2, 30, 37]—with little coverage of motion edits. Second, datasets containing motion edits are small and of low quality: motion cate-

[1] https://github.com/elainew728/motion-edit

[2] https://huggingface.co/datasets/elaine1wan/MotionEdit-Bench

[3] https://huggingface.co/datasets/elaine1wan/MotionEdit-Train

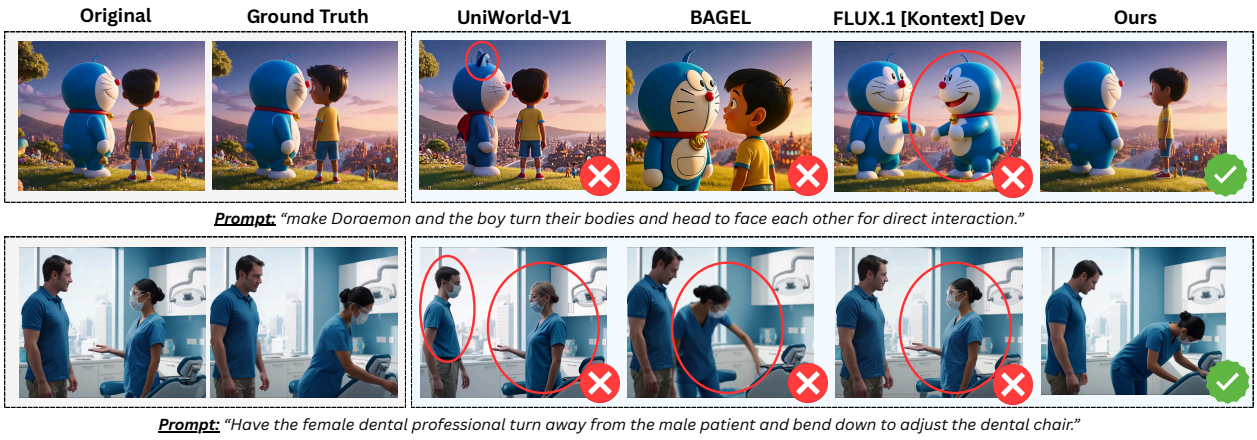

Figure 2. Qualitative comparison of state-of-the-art image editing models on MOTIONEDIT. Existing models fail to execute the required motion edits (e.g. UNIWORLD-V1 fail to edit subject postures and FLUX.1 KONTEXT produces severe identity distortions), while our MotionNFT-trained model accurately performs the intended motion edit that closely matches the ground-truth.

gories are unclear, and the provided target edits are often unfaithful or physically implausible [2, 14, 36]. As shown in Fig. 1, these models frequently fail to achieve intended action changes and introduce visual artifacts, undermining both training supervision and evaluation reliability. These limitations underscore a key challenge in motion image editing: building datasets with precise motion-edit instructions and high-quality, faithful edited targets that preserve appearance and scene context while accurately reflecting the intended action changes.

Motion Estimation in Images. Motion estimation is a long-standing problem in computer vision. Modern approaches rely on optical flow, which predicts per-pixel displacement between two images [9, 25, 27, 32]. Recent work such as UniMatch [33] further advances large-displacement estimation by formulating optical flow as a global matching problem unified with stereo tasks. Inspired by the effectiveness of optical flow in capturing fine-grained motion changes, we propose a motion-centric reward framework based on optical flow, which quantitatively measures how accurately a model performs the intended motion edit in synthesized images.

Reinforcement Learning for Image Generation. Policy-gradient methods such as PPO [21, 23] and GRPO [15, 24] have been explored for improving image generation. More recently, DiffusionNFT [38] introduces negative-aware finetuning, which contrasts positive and negative generations during the forward diffusion process to obtain an implicit policy improvement direction, steering the model toward high-reward outcomes while repelling low-reward ones. UniWorld-V2 [13] extends DiffusionNFT by integrating an MLLM-based online scoring pipeline for rating editing aspects like prompt compliance and style fidelity. However, current RL-based post-training frameworks remain motion-agnostic: they emphasize semantic correctness and visual details, yet offer no supervision on how subjects and objects should *move* for motion-centric edits.

3. Dataset Construction

3.1. Problem Definition and Categorization

The task of motion image editing has not been comprehensively explored in prior works. Therefore, we first provide a systematic definition of this novel task.

Motion Image Editing. Given an input image and a natural-language instruction specifying a target motion change (e.g. *"make the woman drink from the cup"*), the goal is to synthesize an edited image where: (1) the edited motion faithfully reflects the intended action; (2) the resulting pose or interaction is physically plausible and respects articulated constraints (e.g., *"slightly open his eyes"*); (3) non-edited factors like appearance, background, and viewpoint remains consistent. Unlike traditional appearance-focused editing, motion editing requires models to interpret the instructed motion and translate it into coherent spatial changes in the image, requiring fine-grained spatial and kinematic understanding.

3.2. Dataset Construction Pipeline

As discussed in Section 2, existing image editing datasets and benchmarks lack reliable ground-truth targets that correctly execute the instructed motion while preserving subject identity and scene context. Prior datasets either introduce artifacts and hallucinations, alter appearance, or unintentionally shift viewpoint or scale. Sourcing high-quality motion edit ground truth remains a challenging problem. Instead of synthesizing edited targets as in prior work [2, 36], we propose a **video-driven data construction pipeline** that mines paired frames from dynamic video sequences to produce high-quality (input image, edit instruction, target image) triplets. These data reflect naturally occurring and coherent motion transitions grounded in video kinematics. Full details on dataset construction are in the "Additional Dataset Construction Details" Appendix section.

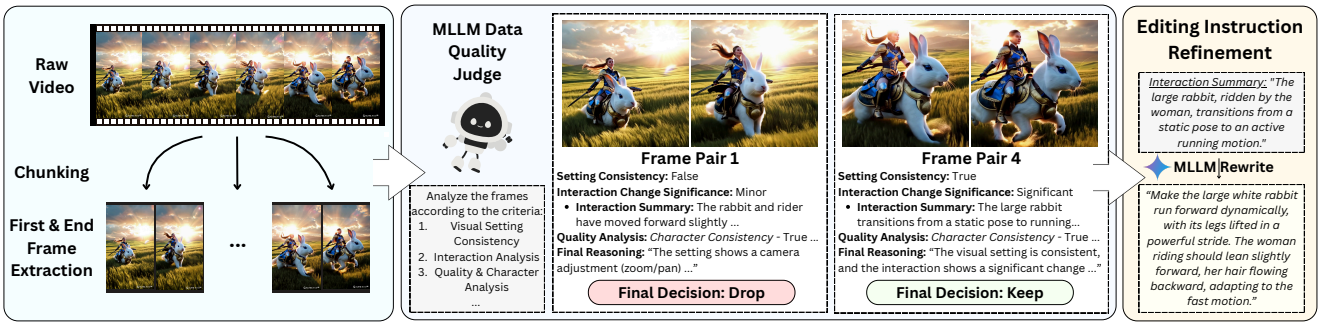

Figure 3. MotionEdit's data construction pipeline. We segment raw videos, extract frame pairs, and automatically filter them using an MLLM data quality judge. For all kept pairs, we use a MLLM rewrite module to generate clean, motion-focused editing instructions. Our pipeline enables scalable construction of high-quality motion editing data and can be extended to much larger video corpora.

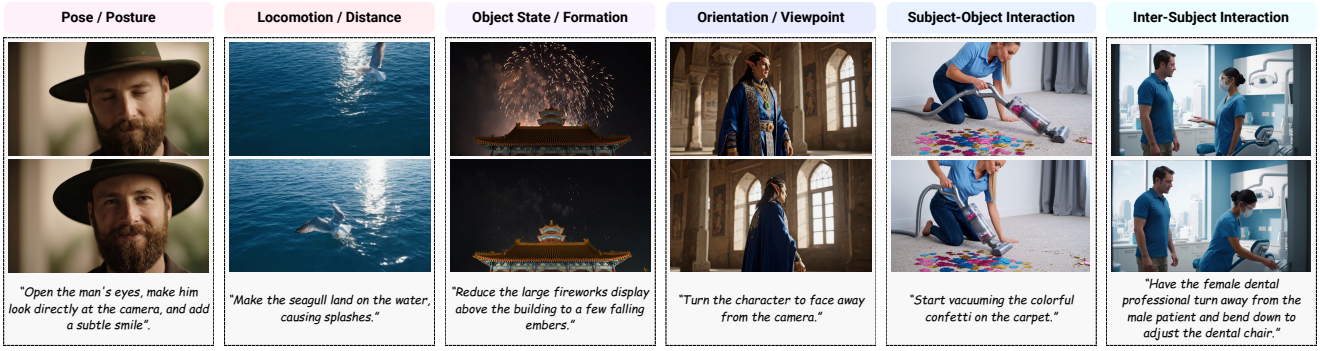

Figure 4. Example categories of data in MOTIONEDIT. Drawing from diverse video sources, our dataset captures a broad spectrum of motion transformations, including pose shifts, locomotion, viewpoint changes, and both subject–object and inter-subject interactions.

Video Collection To obtain frame pairs capturing clean motion transitions, we first explored conventional human action datasets such as HAA500 [5] and K400 [10]. Although diverse, these datasets often suffer from problems like low resolution, motion blur, rapid viewpoint shifts, etc., making them unsuitable for extracting faithful pre-/post-edit pairs that preserve identity and background consistency.

In contrast, recent Text-to-Video (T2V) models (e.g. Veo-3 [7], Kling-AI [11]) produce visually sharp, temporally smooth videos with stable subjects and backgrounds. We therefore draw from two publicly released T2V video collections—ShareVeo3 [28] and the KlingAI Video Dataset [19]—as our initial pool of candidate videos. We then apply further processing to extract high-quality frame pairs for our MOTIONEDIT dataset.

Frame Extraction and Automatic Validation Given the video pool, our goal is to identify frame pairs that exhibit meaningful motion changes while preserving all non-motion factors. We segment each video into 3-second windows and sample the first and last frame of each segment, providing a broad and efficient set of candidate motion transitions. However, many sampled pairs are unusable due to camera motion, subject disappearance, environmental changes, or visual degradation. Motivated by the recent success of LLM/MLLM-based data filtering [3, 4, 8, 29], we leverage Google's Gemini [26] model to automatically filter these cases at scale. We prompt Gemini to evaluate each frame pair along three critical dimensions:

- **Setting Consistency.** Verify that background, viewpoint, and lighting remain stable despite subject motion.
- **Motion and Interaction Change.** Identify interaction states in each frame and summarize the primary motion transition (e.g., "not holding cup → drinking"). The model also judges whether the change is significant enough to constitute a meaningful motion edit.
- **Subject Integrity and Quality.** Ensure the main subjects are present, identifiable, and artifact-free, avoiding cases with occlusion, shrinkage, hallucinations, and distortions.

Based on these criteria, the MLLM outputs a binary keep/discard decision. A pair is accepted only if (1) the scene remains stable, (2) the motion change is non-trivial, (3) subjects are consistent and coherent, and (4) both frames maintain high visual quality. This filtering is essential for obtaining high-quality motion edit triplets for our dataset.

3.3. Editing Prompt Construction

While the validated frame pairs provide reliable visual reference, their corresponding edit instructions must be clear, natural, and semantically faithful to the observed change. We convert the MLLM-generated motion-change summaries into user-style editing prompts by following

the prompt refinement procedure of Wu et al. [31]. This step removes unnecessary analysis details and standardizes prompts into imperative form (e.g. "Make the woman turn her head toward the dog."), ensuring alignment between the described edit and the actual motion transition in data.

3.4. Dataset Statistics

Our final MOTIONEDIT dataset consists of 10,157 motion-editable frame pairs, sourced from both Veo-3 and KlingAI video collections. Specifically, we obtain 6,006 samples from Veo-3 and 4,151 samples from KlingAI. We perform a random 90/10 train-test split, resulting in 9,142 training data and 1,015 evaluation data that constitutest MOTIONEDIT-BENCH. Each sample includes a source or input image, a target image exhibiting a real motion transition from the original video, and a precise motion edit instruction. As shown in Figure 4, data in MOTIONEDIT can be generally categorized into six motion edit types:

- **Pose / Posture**: Changes in body configuration position (e.g. raising hand) while keeping identity and scene fixed.
- **Locomotion / Distance**: Changes in subject's spatial position or distance relative to the camera or environment.
- **Object State / Formation**: Changes in the physical form or condition of an object (e.g., deformation, expansion).
- **Orientation / Viewpoint**: Changes in subject's facing direction or angle without position change.
- **Subject–Object Interaction**: Changes in how a person or agent physically interacts with an object (e.g., holding).
- **Inter-Subject Interaction**: Changes in the coordinated motion between two or more subjects (e.g., facing).

3.5. Data Motion Magnitude Comparison

To quantify and compare the amount of motion present in before-after editing pairs between MOTIONEDIT and other editing datasets, we randomly select 100 data from each dataset and calculate the overall pixel-level motion displacement between each input image and its corresponding edited target. We measure motion changes in the image pairs with optical flow, the calculation of which is explained later in Section 4.

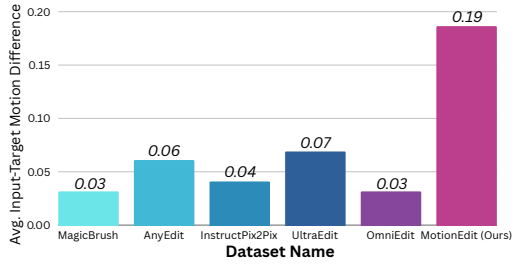

Figure 5. Comparison of motion difference between before- and post-edit images in different datasets [2, 30, 35–37]. Our MOTIONEDIT dataset achieves the most significant motion changes.

Figure 5 reports the average input-target motion magnitude across 6 editing datasets. Prior datasets such as MagicBrush [36], AnyEdit [35], InstructPix2Pix [2], UltraEdit [37], and OmniEdit [30] contain relatively modest motion changes (typically around 0.05), whereas our MOTIONEDIT dataset exhibits substantially larger motion differences (0.19), representing **5.8**× greater motion than MagicBrush and OmniEdit and **3**× that of UltraEdit. This highlights our contribution of a challenging motion editing dataset with substantial motion transformation.

4. Learning Motion Image Editing

4.1. Preliminaries

Flow Matching Models Recent progress in diffusion models has shifted from Denoising Diffusion Probabilistic Models (DDPMs) [22] to Flow Matching Models (FMMs) [12]. Given noisy sample z_t and conditioning c, FMMs reformulate the noise prediction process in DDPMS by estimates a deterministic *velocity field* v that transports z_t toward its clean counterpart. As a result, inference for FMMs reduces to the ODE $dz_t = v_\theta(z_t, t, c)\,dt$, which enables efficient generation compared to DDPM sampling.

Diffusion Negative-aware Finetuning (NFT) DiffusionNFT [38] enhances FMM reward training by learning not only a *positive* velocity $v^+(x_t, c, t)$ that the model should move toward, but also a *negative* velocity $v^-(x_t, c, t)$ that it should avoid. The training objective is:

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

where v is the target velocity and v_θ^+, v_θ^- are defined as linear combinations of the old and current policies:

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

A key challenge is obtaining a calibrated reward r that accurately reflects whether a sample should be treated as "positive". Since raw rewards may differ in scale or distribution, DiffusionNFT transforms them into an *optimality reward*:

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

where Z_c is a normalization factor (e.g., the global reward standard deviation). This normalization stabilizes learning and ensures consistent positive/negative assignment across prompts and reward models.

4.2. MotionNFT: Motion-Aware Reward for NFT

We introduce **MotionNFT**, a motion-aware reward framework designed for NFT training on motion-editing tasks. Since our objective is to evaluate how accurately a model applies the intended action to subjects and objects, our reward function must quantify the alignment between model-predicted motion and the ground-truth motion edit. Inspired

Model	MotionEdit-Bench					
	Overall↑	Fidelity↑	Preservation↑	Coherence↑	Motion Alignment Score (MAS)↑	Win Rate↑
Instruct-P2P [2]	1.30	1.32	1.29	1.29	34.15	16.09
AnyEdit [35]	1.31	1.32	1.32	1.30	35.11	16.88
MagicBrush [36]	1.50	1.58	1.47	1.44	44.24	19.51
UltraEdit [37]	2.42	1.88	2.09	2.13	47.18	28.33
UniWorld-V1 [13]	2.87	2.96	2.76	2.88	55.37	41.14
Step1X-Edit [16]	4.02	4.04	3.99	4.02	52.98	61.14
BAGEL [6]	4.10	4.24	4.01	4.06	51.83	61.46
FLUX.1 Kontext [Dev] [12]	3.84	3.89	3.79	3.83	53.73	57.71
+MotionNFT *(Ours)*	**4.25**	**4.33**	**4.16**	**4.25**	**55.45**	**64.95**
Qwen-Image-Edit [31]	4.65	4.70	4.59	4.66	56.46	72.80
+MotionNFT *(Ours)*	**4.72**	**4.79**	**4.63**	**4.74**	**57.23**	**73.67**

Table 1. Quantitative results on MotionEdit-Bench. Among existing methods, Step1X-Edit and BAGEL achieve the strongest motion-editing performance, while diffusion-based editors such as AnyEdit and MagicBrush perform poorly across both generative and discriminative metrics. FLUX.1 Kontext and Qwen-Image-Edit models trained with MotionNFT yields the best overall results: for both models, applying MotionNFT boosts all generative metrics, MAS and pairwise win rate.

by the use of optical flow for measuring motion between consecutive video frames, we adopt an optical-flow–based **motion-centric scoring framework** that treats each input–edit pair as an implicit "before–after" sequence.

Given a triplet $\mathbf{X} = (\mathbf{I}_{\text{orig}}, \mathbf{I}_{\text{edited}}, \mathbf{I}_{\text{gt}})$, we compute optical flow fields using a pretrained estimator [33]. The predicted motion is $\mathbf{V}_{\text{pred}} = \mathcal{F}(\mathbf{I}_{\text{orig}}, \mathbf{I}_{\text{edited}})$ and the ground-truth motion is $\mathbf{V}_{\text{gt}} = \mathcal{F}(\mathbf{I}_{\text{orig}}, \mathbf{I}_{\text{gt}})$, where each flow lies in $\mathbb{R}^{H\times W\times 2}$. We normalize both flows by the image diagonal to ensure scale consistency across resolutions.

Motion magnitude consistency. We measure the deviation between flow magnitudes using a robust ℓ_1 distance: $\mathcal{D}_{\text{mag}} = \frac{1}{HW}\sum_{i,j}(\|\tilde{\mathbf{V}}_{\text{pred}}(i,j) - \tilde{\mathbf{V}}_{\text{gt}}(i,j)\|_1 + \varepsilon)^q$, where $q \in (0,1)$ is a constant term to suppress outliers.

Motion direction consistency. We compute cosine-based directional error between the unit flow vectors $e_{\text{dir}}(i,j) = \frac{1}{2}\big(1 - \hat{\mathbf{v}}_{\text{pred}}(i,j)^\top \hat{\mathbf{v}}_{\text{gt}}(i,j)\big)$, and weight each pixel by its relative ground-truth motion magnitude. The directional misalignment is $\mathcal{D}_{\text{dir}} = \frac{\sum_{i,j} w(i,j) e_{\text{dir}}(i,j)}{\sum_{i,j} w(i,j)+\varepsilon}$.

Movement regularization. To prevent trivial edits that make almost no motion, we compare the average predicted and ground-truth magnitudes: $M_{\text{move}} = \max\{0,\ \tau + \frac{1}{2}\bar{m}_{\text{gt}} - \bar{m}_{\text{pred}}\}$, where τ is a small positive margin and $\bar{m}$ denotes the spatial mean.

Combined reward. We aggregate the three terms into a composite distance $\mathcal{D}_{\text{comb}} = \alpha\,\mathcal{D}_{\text{mag}} + \beta\,\mathcal{D}_{\text{dir}} + \lambda_{\text{move}} M_{\text{move}}$ where α, β, and λ_{move} are constants that balances term scales and weightings. The composite distance is then normalized and clipped: $\tilde{D} = \text{clip}\left((\mathcal{D}_{\text{comb}} - \mathcal{D}^*_{\text{min}})/(\mathcal{D}_{\text{max}} - \mathcal{D}^*_{\text{min}}), 0, 1\right)$, and converted into a continuous reward $r_{\text{cont}} = 1 - \tilde{D}$. Finally, we quantize it into 6 discrete reward levels: $r_{\text{motion}} = \frac{1}{5}\,\text{round}(5\,r_{\text{cont}}) \in \{0.0,\ 0.2,\ 0.4,\ 0.6,\ 0.8,\ 1.0\}$. The resulting scalar reward is used to compute optimality rewards and update the policy model v_θ under the DiffusionNFT objective (Eq. 1). Figure 6 illustrates the MotionNFT reward pipeline.

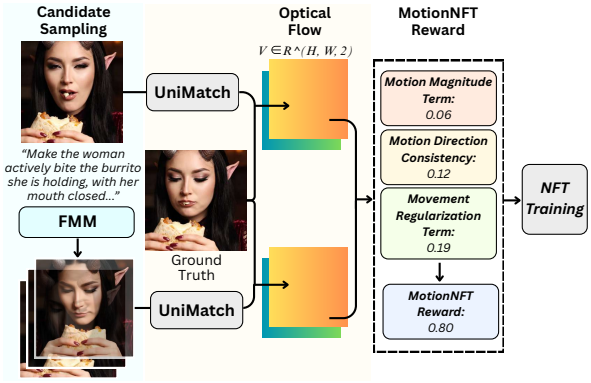

Figure 6. MotionNFT's Reward Scoring pipeline. For each sampled model-edited image, we measure the alignment between the input-generated optical flow and the input-ground truth optical flow, obtaining the final reward score.

5. Experiments

5.1. Experimental Setup

We provide important details of our experimental setups. Full details are reported in the *Additional Experiment Details* Appendix section.

MotionNFT Training We use FLUX.1 Kontext [Dev] [12] and Qwen-Image-Edit [31] as base models for MotionNFT training. Following Lin et al. [13]'s implementation, we use Fully Sharded Data Parallelism (FSDP) for text encoder and apply gradient checkpointing in training for GPU memory usage optimization. To improve models' motion image editing capabilities while preserving general image editing ability, we employ a multi-score reward formulation with a weighted combination of (*i*) 50% our optical flow-based *Motion Reward* r_motion and (*ii*) 50% MLLM reward proposed by Lin et al. [13]. For

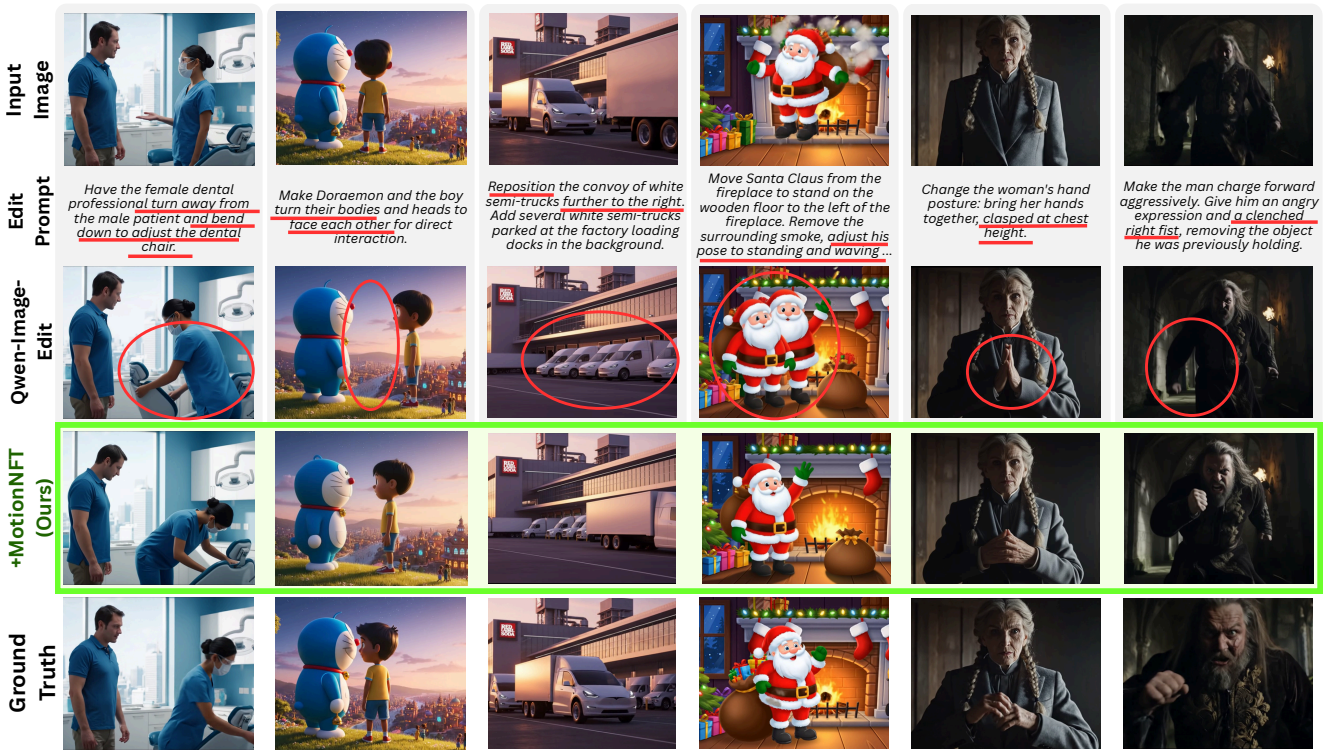

Figure 7. Qualitative examples of our MotionNFT. The baseline QWEN-IMAGE-EDIT [31] model often fails to execute the instructed motion (circled regions), producing edits that do not match the required action change (red underlines). With MotionNFT training, the model succeeds in performing precise motion edits that closely align with the ground-truth transformations.

MLLM-based evaluation, we serve a QWEN2.5-VL-32B-INSTRUCT [1] model via vLLM on a separate node that performs online scoring throughout training. The optical flow component of our reward leverages a lightweight UniMatch model (335.6M parameters), which we run directly on the training nodes to provide efficient motion-level guidance.

Benchmarked Image Editing Models We evaluate 9 open-source models on MOTIONEDIT-BENCH: Instruct-P2P [2], MagicBrush [36], AnyEdit [35], UltraEdit [37], Step1X-Edit [16], BAGEL [6], UniWorld-V1 [13], FLUX.1 Kontext [Dev] [12], and Qwen-Image-Edit [31].

5.2. Evaluation Metrics

Generative Metrics. Following Luo et al. [17] and Lin et al. [13], we use an MLLM to evaluate edited images with four generative metrics: *Fidelity*, *Preservation*, *Coherence*, and their *Overall* average. We choose to use Google's Gemini [26] as the MLLM evaluator and use evaluation prompts adapted from the "action" category of Luo et al. [17].

Discriminative Motion Alignment Score (MAS). To complement the MLLM generative metric with deterministic assessment, we introduce an optical flow–based Motion Alignment Score (MAS) to measure how well the model understands and performs the correct motion change in images. MAS combines the *motion magnitude consistency* term $\mathcal{D}_{\text{mag}}$ and the *motion direction consistency* term $\mathcal{D}_{\text{dir}}$ from Section 4 into a single motion alignment metric: $\mathcal{D}_{\text{ovl}} = \alpha\,\mathcal{D}_{\text{mag}} + (1-\alpha)\,\mathcal{D}_{\text{dir}}$, where α is a constant term balancing scales. Then, we normalize $\mathcal{D}_{\text{ovl}}$ and convert it into: $\text{MAS} = 100.00 \cdot (1 - \text{clip}\left((\mathcal{D}_{\text{ovl}} - d_{\min})/(d_{\max} - d_{\min}), 0, 1\right))$. Higher scores indicate closer alignment. If the predicted motion is nearly static compared to ground truth, i.e., $\mathbb{E}[m_{\text{pred}}]/\mathbb{E}[m_{\text{gt}}] < \rho_{\min}$, we assign $\text{MAS} = 0$.

Pairwise Win Rate We additionally compute pairwise win rate between models on the same evaluation data based on overall MLLM scores. We define the pairwise win rate as $(\text{wins} + 0.5 \cdot \text{draws})/\text{total}$, and report each model's mean win rate averaged over all comparisons with other models.

5.3. Quantitative Evaluation Results

Table 1 reports quantitative performance of 9 image editing models on MOTIONEDIT-BENCH. The first 4 columns shows MLLM generative ratings on a 0–5 scale. Our optical flow-based *MAS* metric measures motion consistency on a 0–100 scale. The *Win Rate* reflects the percentage of pairwise comparisons in which a model's output received a higher average MLLM score than a competing one.

#1: Improved Motion Editing Quality. Across both base models, MotionNFT consistently improves all aspects of generation quality on motion editing, as measured by the generative evaluator. When applied to FLUX.1 KONTEXT, MotionNFT raises the Overall score from 3.84 to

4.25 (+10.68%), with notable gains in Fidelity (+0.44) and Coherence (+0.42). For QWEN-IMAGE-EDIT, MotionNFT also improves the Overall score from 4.65 to 4.72.

#2: Enhanced Motion Alignment. MotionNFT yields substantial improvements in MAS, highlighting its effectiveness in producing motion changes consistent with the ground-truth edits. On FLUX.1 KONTEXT, MotionNFT increases MAS from 53.73 to 55.45, while on QWEN-IMAGE-EDIT, MAS improves from 56.46 to 57.23. These gains are achieved despite the strong baselines and show that our flow-based reward provides meaningful guidance for learning spatial and motion-aware transformations.

#3: Strong Pairwise Preference Performance. MotionNFT also achieves higher win rates relative to all evaluated models. For FLUX.1 KONTEXT, MotionNFT boosts win rate from 57.97% to 65.16% (+12.40%), and from 72.99% to 73.87% for QWEN-IMAGE-EDIT. These results show that MotionNFT produces more accurate motion edits that are more frequently preferred over outputs of other models.

5.4. Qualitative Evaluation Results

Figure 2 and Figure 7 illustrate representative qualitative results on MOTIONEDIT.

 # 1: Existing models struggle to perform correct motion edits. We observe that even state-of-the-art open-sourced image editing models like FLUX.1 Kontext and Qwen-Image-Edit struggle to correctly perform motion-centric changes like turning body directions. These models often preserve the original pose or only apply superficial appearance changes. This highlights the crucial bottleneck in translating motion-related language instructions into coherent image subject / object transformations.

#2: MOTIONNFT improves motion editing capability. Training with MOTIONNFT enables Qwen-Image-Edit to produce outputs that more faithfully reflect the intended motion, e.g. rotating character directions, adjusts limb and torso positions to reflect bending or turning actions. Additionally, the resulting edits preserve identity and scene context while achieving the targeted motion change, closely matching the ground-truth transformations. These observations validates the effectiveness of incorporating motion-centric guidance in MotionNFT to execute meaningful, structure-aware motion edits that current image editing models consistently fall short in achieving.

5.5. Ablation Studies

General Image Editing Performance To verify that MotionNFT preserves a model's general editing ability, we follow previous work [13] and conduct evaluation on ImgEdit-Bench [34], a comprehensive benchmark covering 8 editing subtasks. Table 2 shows that MotionNFT consistently improves or maintains performance across all categories for both FLUX.1 KONTEXT and QWEN-IMAGE-EDIT, even yielding higher overall scores. Results confirm that MotionNFT can enhance models' motion editing performance without trading off general editing quality.

Model	ImgEdit-Bench								
	Add	Adj.	Rpl.	Rem.	Bck.	Stl.	Hyb.	Act.	Ovl.↑
FLUX.1 Kontext	3.54	2.90	3.73	2.89	3.59	3.96	2.90	2.56	3.26
+ MOTIONNFT	**3.71**	**3.28**	**3.93**	**3.05**	**3.72**	**4.41**	**2.99**	**2.85**	**3.50**
Qwen-Image-Edit	4.20	3.70	4.22	4.20	4.17	4.60	3.55	**4.03**	4.08
+ MOTIONNFT	**4.31**	**3.72**	**4.46**	**4.30**	**4.21**	**4.67**	**3.96**	3.87	**4.20**

Table 2. Results on ImgEdit-Bench [34] MotionNFT not only preserves, but oftentimes boosts general editing performances.

Model	MotionEdit-Bench		
	Overall. ↑	MAS ↑	Win Rate ↑
FLUX.1 Kontext	3.84	53.73	57.97
+ *UniWorld-V2[13]*	4.20	54.58	64.02
+MOTIONNFT *(Ours)*	**4.25**	**55.45**	**65.16**
Qwen-Image-Edit	4.65	56.46	73.01
+ *UniWorld-V2[13]*	4.70	56.46	72.77
+MOTIONNFT *(Ours)*	**4.72**	**57.23**	**73.87**

Table 3. Comparison to training with MLLM-based reward [13] only. Incorporating MotionNFT yields noticeable improvements MLLM-scored Overall editing quality, optical flow-based Motion Alignment Score, and the pairwise Win Rate across all models.

Comparison with MLLM-only Reward To verify the effect of MotionNFT's supervision, we compare MotionNFT against the MLLM-only RL framework in UniWorld-V2 [13]. Table 3 shows that while MLLM-only training yields modest improvements over the base models, MotionNFT consistently achieves higher overall edit quality, better motion alignment, and superior win rates across both base models. These results demonstrate that incorporating optical flow-based motion guidance yields more targeted and effective motion-editing capabilities.

6. Conclusion

We introduced MOTIONEDIT, a high-quality dataset and benchmark for the novel motion image editing task, aiming at correct modifying subject actions and interactions in images while preserving identity and scene consistency. To improve model performance on this challenging task, we proposed MOTIONNFT, a motion-guided negative-aware finetuning framework that integrates an optical-flow–based motion reward for training. MotionNFT provides supervision on motion magnitude and direction, enabling models to understand and perform motion transformations that existing models consistently struggle with. Both quantitative and qualitative experiment results demonstrate that MotionNFT delivers consistent gains across generative quality, motion alignment, and preference win rate on two strong base models, FLUX.1 Kontext and Qwen-Image-Edit.

References

[1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 7

[2] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. *arXiv preprint arXiv:2211.09800*, 2022. 2, 3, 5, 6, 7

[3] Derin Cayir, Renjie Tao, Rashi Rungta, Kai Sun, Sean Chen, Haidar Khan, Minseok Kim, Julia Reinspach, and Yue Liu. Refine-n-judge: Curating high-quality preference chains for llm-fine-tuning. *arXiv preprint arXiv:2508.01543*, 2025. 4

[4] Ruibo Chen, Yihan Wu, Lichang Chen, Guodong Liu, Qi He, Tianyi Xiong, Chenxi Liu, Junfeng Guo, and Heng Huang. Your vision-language model itself is a strong filter: Towards high-quality instruction tuning with data selection. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 4156–4172, 2024. 4

[5] Jihoon Chung, Cheng hsin Wuu, Hsuan ru Yang, Yu-Wing Tai, and Chi-Keung Tang. Haa500: Human-centric atomic action dataset with curated videos. In *ICCV 2021*. 4

[6] 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. 2, 6, 7

[7] Google DeepMind. Veo 3. https://deepmind.google/models/veo/, 2025. Accessed: 2025-11. 4

[8] Erik Henriksson, Otto Tarkka, and Filip Ginter. Finerweb-10bt: Refining web data with llm-based line-level filtering. In *Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025)*, pages 258–268, 2025. 4

[9] Shihao Jiang, Dylan Campbell, Yao Lu, Hongdong Li, and Richard Hartley. Learning to estimate hidden motions with global motion aggregation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9772–9781, 2021. 2, 3

[10] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 4

[11] Kuaishou Technology. Kling ai. https://app.klingai.com/global/image-to-video/, 2025. Accessed: 2025-11. 4

[12] 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. 2, 5, 6, 7

[13] 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, 6, 7, 8

[14] Haonan Lin, Yan Chen, Jiahao Wang, Wenbin An, Mengmeng Wang, Feng Tian, Yong Liu, Guang Dai, Jingdong Wang, and Qianying Wang. Schedule your edit: A simple yet effective diffusion noise schedule for image editing. *Advances in Neural Information Processing Systems*, 37:115712–115756, 2024. 3

[15] Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. *arXiv preprint arXiv:2505.05470*, 2025. 3

[16] 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. 6, 7

[17] 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. 7

[18] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*. 2

[19] Nyuuzyou. Klingai video dataset. https://huggingface.co/datasets/nyuuzyou/klingai, 2025. Accessed: 2025-05. 4

[20] OpenAI. Image generation api, 2025. https://openai.com/index/image-generation-api/. 2

[21] Allen Z Ren, Justin Lidard, Lars Lien Ankile, Anthony Simeonov, Pulkit Agrawal, Anirudha Majumdar, Benjamin Burchfiel, Hongkai Dai, and Max Simchowitz. Diffusion policy policy optimization. In *CoRL 2024 Workshop on Mastering Robot Manipulation in a World of Abundant Data*. 3

[22] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, 2022. 5

[23] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. 3

[24] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. 3

[25] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8934–8943, 2018. 2, 3

[26] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 4, 7

[27] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pages 402–419. Springer, 2020. 2, 3

[28] Wenhao Wang and Yi Yang. Vidprom: A million-scale real prompt-gallery dataset for text-to-video diffusion models. 2024. 4

[29] Weizhi Wang, Yu Tian, Linjie Yang, Heng Wang, and Xifeng Yan. Open-qwen2vl: Compute-efficient pre-training of fully-open multimodal llms on academic resources. *arXiv preprint arXiv:2504.00595*, 2025. 4

[30] Cong Wei, Zheyang Xiong, Weiming Ren, Xinrun Du, Ge Zhang, and Wenhu Chen. Omniedit: Building image editing generalist models through specialist supervision. *arXiv preprint arXiv:2411.07199*, 2024. 2, 5

[31] 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. 2, 5, 6, 7

[32] Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, and Dacheng Tao. Gmflow: Learning optical flow via global matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8121–8130, 2022. 2, 3

[33] 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. 2, 3, 6

[34] Yang Ye, Xianyi He, Zongjian Li, Bin Lin, Shenghai Yuan, Zhiyuan Yan, Bohan Hou, and Li Yuan. Imgedit: A unified image editing dataset and benchmark. *arXiv preprint arXiv:2505.20275*, 2025. 8

[35] 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. 5, 6, 7

[36] Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing. In *Advances in Neural Information Processing Systems*, 2023. 2, 3, 5, 6, 7

[37] 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. 2, 5, 6, 7

[38] 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. 2, 3, 5