KINESYS: TEST-TIME ADAPTATION FOR REFERENCE-BASED EFFECT EDITING IN IMAGES

Anonymous authors
Paper under double-blind review

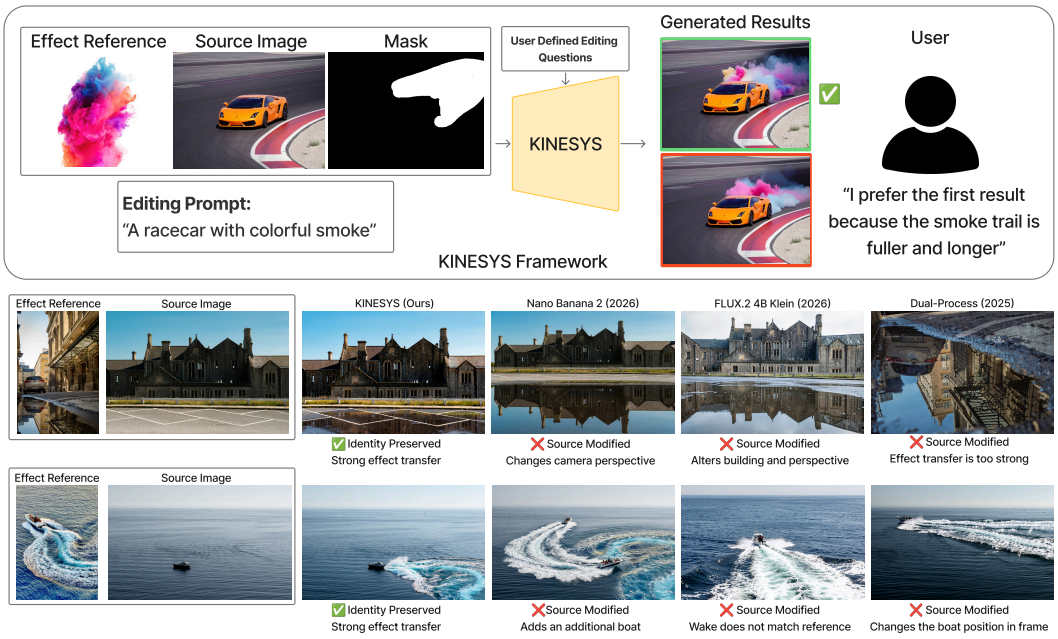

Figure 1: We present KINESYS, a test-time adaptation framework for reference-based visual effects editing in images. (Top) The artist supplies a source, effect reference, optional mask, editing prompt, and verification questions, and KINESYS generates multiple samples with diverse styles. (Bottom) KINESYS transfers puddle and boat-wake effects while preserving source composition and identity; baselines change the scene, add unwanted content, or miss effect properties.

ABSTRACT

Visual effects are often easier to demonstrate via reference than to describe in words. We study effect-reference editing, where a reference image shows a desired visual effect and language identifies which properties should guide the edit. Existing single-pass editing techniques often fail to make fine-grained edits, resulting in subtle identity distortions or deviations from the reference effect. We introduce KINESYS, a test-time adaptation framework for effect-reference editing that combines reference-aware and source-aware feedback to transfer the desired effect while preserving the identity of the original shot. To ensure KINESYS can produce a diverse set of identity-preserving edits, we develop a robust optimization strategy in which the generative model is fine-tuned across multiple noise samples and optimization runs. Evaluation with VLM judges, source-preservation metrics (LPIPS, TPIPS), and a user study shows that KINESYS outperforms state-of-the-art image editors in source identity preservation; on effect transfer, it remains competitive with Nano Banana 2 and outperforms open-weight editors.

1 INTRODUCTION

Recent open-weight and proprietary image editors can make convincing changes to existing images (Black Forest Labs, 2026; Qwen Team, 2025; Raisinghani, 2026; OpenAI, 2026; Team See-

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

dream et al., 2025). Editing visual effects nevertheless exposes a difficult control problem: artists often need to add elements such as smoke, fire, or rain, or introduce visible damage, without unintended changes to the source subject's identity, camera angle, framing, or surrounding scene. An instruction-based edit such as "add tire smoke behind the car" names the new element but does not specify its texture, color, scale, density, placement, or interaction with the road and vehicle. A visual reference can supply these details, but the editor must separate the properties that define the desired effect from incidental reference content. The central challenge is therefore to realize the requested effect strongly enough while preserving the source subject and scene structure.

We study *effect-reference editing*: the user provides a source image, an effect reference, and a text instruction identifying which visual and spatial properties should guide the edit. The user specifies which effect properties to transfer through reference-grounded yes-or-no questions and may include a mask to localize the edit. The effect reference specifies what to introduce, while the source defines the subject identity and scene structure that should be preserved. Modern in-context editors can accept source images, references, and language through a common interface (Black Forest Labs, 2026; Team Seedream et al., 2025), making them a natural basis for this setting.

A single editing pass may under-edit, producing a weak or generic result, or follow the reference too closely and alter the subject, viewpoint, or scene layout, as illustrated in Figure 1. A natural response to single-pass failures is to evaluate candidate edits and use that feedback to improve subsequent outputs. KINESYS fine-tunes the generator using reference-grounded visual question answering (VQA): a vision-language model (VLM) answers the supplied questions about each candidate. These judges provide different information: our Gemma-based VQA judge (Google DeepMind, 2026) sees the reference but not the source and assesses the selected effect properties, while EditReward (Wu et al., 2026b) sees the source but not the reference and assesses instruction following and visual quality. We combine these judges to support identity-preserving effect transfer.

However, optimizing the generator with complementary feedback remains challenging. The same editing request can produce different results depending on the random noise used to generate its candidates (Ma et al., 2025; Guimaraes & Perona, 2026; Ben-Hamu et al., 2024; Kim et al., 2025). Because these candidates also provide the feedback for fine-tuning, an unlucky initial set can steer the generator toward a poor solution. To reduce this dependence, KINESYS starts K independent fine-tuning runs, each using N fixed-noise candidates, then prunes runs and selects a checkpoint.

Together, these design choices adapt KINESYS to one request, allowing it to produce several identity-preserving versions of the referenced effect from unseen noise samples. As Figure 7 shows, new noise changes details such as texture, shape, density, and placement, giving artists several interpretations of the same effect to compare.

We evaluate KINESYS on 81 editing requests spanning eight visual effects. KINESYS achieves the strongest source-preservation scores among the evaluated methods, remains competitive with Nano Banana 2 on editing quality, and leads the evaluated open-weight editors on both source preservation and editing quality. The blinded user study shows that participants selected KINESYS more often for identity preservation against Nano Banana 2 (Raisinghani, 2026) (45.3% versus 24.7%) and Step1X-Edit (Liu et al., 2025a) (53.6% versus 20.9%), while effect-transfer preferences were competitive with Nano Banana 2 and favored KINESYS over Step1X-Edit.

Our contributions are threefold:

- We introduce KINESYS, a test-time generator adaptation framework for reference-based visual effect editing in images that uses reference-grounded and source-aware VLM feedback to perform identity-preserving effects editing without effect-specific training data.
- We develop a robust optimization strategy for fine-tuning the generator with LoRA using N fixed-noise candidates per run and pruning across K independent runs, with an early diversity penalty to discourage candidate collapse.
- We create a benchmark of 81 editing requests across eight effects with VLM and human judges measuring editing quality for effect transfer and source identity preservation. Our method outperforms the evaluated open-weight models and surpasses Nano Banana 2 in identity preservation while remaining competitive on effect transfer.

2 Related Work

Dataset-trained and reference-guided image editing. Image editors learn shared behavior from paired or synthetic data, as in InstructPix2Pix (Brooks et al., 2023). Paint by Example and MimicBrush transfer reference content into source regions (Yang et al., 2022; Chen et al., 2024); FreeEdit combines reference and language for mask-free editing (He et al., 2024); and Visual Instruction Inversion infers an operation from a before-and-after pair (Nguyen et al., 2023). Recent image editors and unified multimodal models condition image generation or editing on images and language (Labs et al., 2025; Black Forest Labs, 2026; Wu et al., 2025; Zhang et al., 2025; Team Seedream et al., 2025; Jiang et al., 2026; Diao et al., 2026). KINESYS adapts the generator to one effect-reference request without an effect-specific training set or target edited image.

Generative visual effects. Recent visual-effects generation and compositing work includes MagicVFX, VFX Creator, Omni-Effects, and Over++ (Guo et al., 2024; Liu et al., 2025b; Mao et al., 2025b; Qi et al., 2025). EffectMaker learns reference-based VFX generation from a video dataset, while P-Flow refines prompts against a reference at test time with a fixed video generator (Yang et al., 2026; Zhao & Shou, 2026). KINESYS instead adapts a generator for one source–reference image edit, with the reference defining the effect and the source defining what stays unchanged.

Request-specific and feedback-driven adaptation. Imagic and UniTune (Kawar et al., 2023; Valevski et al., 2023) fine-tune a diffusion model for one source image and target text. DRaFT directly optimizes diffusion parameters against a reward, while HIVE, InstructRL4Pix, CogniEdit, and NP-Edit train shared editors with feedback or dataset supervision (Clark et al., 2024; Zhang et al., 2024; Li et al., 2024; 2025; Kumari et al., 2026). SPIE uses AI feedback for concept-specific adaptation from a small set of reference images (Benarous et al., 2025).

Most closely related, Dual-Process uses a VLM to guide request-specific fine-tuning of an image generator (Luo et al., 2025). It asks whether generated images satisfy a control question, then uses the answer likelihood to update LoRA weights on the generator. This provides a way to learn a visual control without a task-specific training set. KINESYS extends this VLM-guided adaptation to effect-reference editing, with the source image constraining subject identity and scene composition.

Complementary feedback for reference-guided editing. HIVE, ADIEE, EditScore, and EditReward evaluate an edit against its source and instruction without an effect reference (Zhang et al., 2024; Chen et al., 2025; Luo et al., 2026; Wu et al., 2026b). Reference-aware evaluators include ICE-Bench, which judges the instruction, source, reference, and result, and EditCLIP, which compares an edit with a paired reference transformation (Pan et al., 2025; Wang et al., 2025). EVR uses multidimensional reference-aware feedback for dataset-level post-training of multi-reference editors (Miao et al., 2026). KINESYS combines both feedback roles within request-specific fine-tuning.

Reward-guided inference and test-time scaling. Inference-time scaling commonly holds the generator fixed. Best-of-N and adaptive pruning search over noise inputs or partial candidates (Ma et al., 2025; Guimaraes & Perona, 2026; Qu et al., 2026; Yu et al., 2026); D-Flow, RewardFlow, FMTT, trajectory control, and closed-loop generation optimize noise, sampling dynamics, or latent controls (Ben-Hamu et al., 2024; Susladkar et al., 2026; Sabour et al., 2025; Chang et al., 2026; Zhao et al., 2026); and iterative editors revise outputs over multiple rounds (Bai et al., 2025; Zeng et al., 2026). KINESYS instead allocates compute across independent fine-tuning runs and returns an editor that can produce several versions of the referenced effect while preserving the source.

3 Method

An effect reference supplies appearance cues, but it also contains subjects and scene details that should not enter the source. KINESYS addresses this by adapting an image editor for each request using both reference-aware and source-aware feedback. The inputs are a source image, an effect reference, a text instruction, reference-grounded questions, and an optional spatial mask. The reference shows the desired effect on different content; the user chooses which visual and spatial properties to transfer and specifies them through the instruction and reference-grounded questions. Without a mask, the entire source image may be edited. The adapted editor can then produce multiple versions of the effect from different noise samples while preserving the source content.

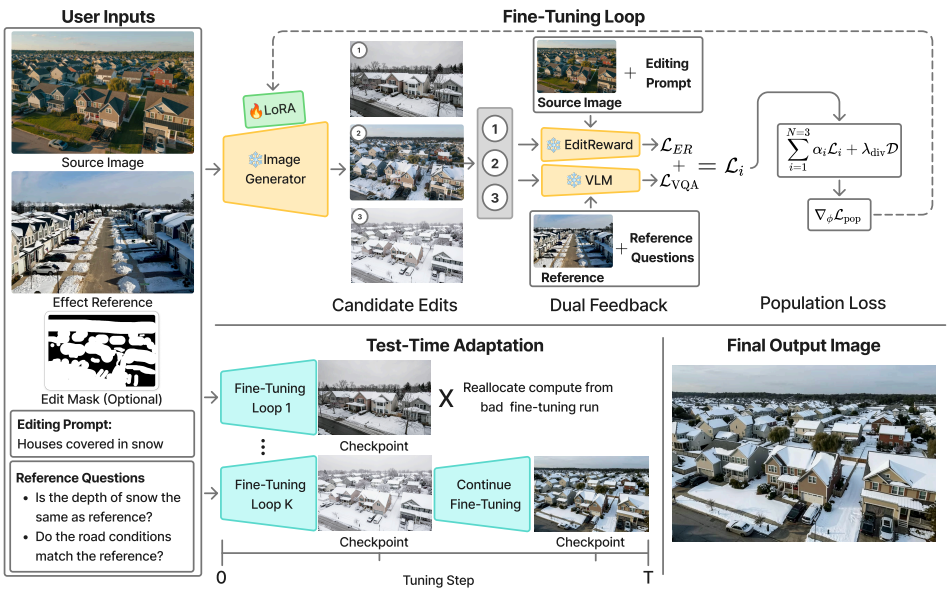

Figure 2: Overview of KINESYS. Reference-grounded VQA and source-aware EditReward provide complementary feedback. Each of K independent runs has its own LoRA parameters ϕ_k and N fixed-noise candidates. Within a run, α_i weights candidate i's objective, while the decaying term $\lambda_{\text{div}}(t)\mathcal{D}$ discourages mode collapse. The 0–T axis marks optimization steps; less promising runs stop at pruning points, while survivors continue to later checkpoints.

Existing editors can produce convincing image edits, but on effect-reference requests they can miss selected effect properties or alter the source subject and scene, as Figure 1 illustrates. We propose KINESYS, a test-time adaptation strategy where VLM judges evaluate effect transfer and preservation of the source image structure and provide feedback to update the generative model.

At each fine-tuning step, KINESYS renders N candidate edits from different fixed noise samples. Reference-grounded VQA and EditReward evaluate each candidate. KINESYS uses their feedback to update only the LoRA weights ϕ (Hu et al., 2021). The base editor, text encoder, and feedback models remain fixed. Independent fine-tuning runs explore different noise sets before the final adapted LoRA weights are selected. Figure 2 illustrates the KINESYS framework.

3.1 COMPLEMENTARY EDITING FEEDBACK

Following Dual-Process (Luo et al., 2025), a frozen vision-language model compares the effect reference r with each candidate edit y_i and answers targeted questions about properties such as the appearance, placement, or interaction of the introduced element. For each of the Q questions q_ℓ, the desired answer a_ℓ is `Yes`. The VQA loss averages the negative log-likelihood of these answers, so lower loss indicates clearer realization of the selected effect properties.

VQA compares each candidate edit with the effect reference but does not see the source, so it cannot directly assess unwanted changes to the original subject or scene. We therefore also use EditReward (Wu et al., 2026b), a learned reward model that sees the source, instruction, and candidate edit. It predicts two scores: instruction following $R_{i,\text{follow}}$ and visual quality $R_{i,\text{quality}}$. The first judges whether an edit is accurate, complete, and limited to the request; the second judges realism and visible artifacts. We average these model outputs as $R_{\text{ER},i} := (R_{i,\text{follow}} + R_{i,\text{quality}})/2$ and minimize $\mathcal{L}_{\text{ER},i} := -R_{\text{ER},i}$. The per-candidate objective combines VQA and EditReward:

$$\mathcal{L}_{\text{VQA},i} = -\frac{1}{Q}\sum_{\ell=1}^{Q} \log p_{\text{VLM}}(a_\ell \mid r, y_i, q_\ell) \quad (1) \qquad \mathcal{L}_i = \lambda_{\text{VQA}}\mathcal{L}_{\text{VQA},i} + \lambda_{\text{ER}}\mathcal{L}_{\text{ER},i} \quad (2)$$

Because optimization can exploit blind spots in learned rewards, we use these two complementary objectives: VQA should strengthen transfer of the selected effect, while EditReward should limit identity drift and visual artifacts. The ablations in Sec. 4.3 and Fig. 5 examine these expectations.

3.2 Robust Optimization

Learning from several candidate edits. A single noise sample can produce a weak edit or one with identity drift, and fine-tuning only that candidate risks adapting the editor around a poor starting point. KINESYS therefore learns from N candidate edits within each run. At the start of the run, it draws N noise samples ϵ_i and reuses them throughout fine-tuning. At each step $t \in \{1, \ldots, T\}$, the current editor renders one candidate from each sample. Reusing the noise makes successive steps comparable and lets feedback track each candidate's progress as the editor changes.

Reward-based candidate weighting. Within a run, all candidates shape the same LoRA. Equal weights would give an edit with artifacts or identity drift as much influence as a stronger edit. We smooth each EditReward score over optimization steps, standardize it across the N candidates, and average the two scores per candidate. A softmax across these averages yields weights α_i that sum to one. This de-emphasizes poorly scored edits without discarding them; weights are recomputed between updates, then detached and held fixed during each LoRA update. The supplementary material gives the smoothing, normalization, and weighting equations.

Maintaining candidate diversity. An artist editing smoke around a car may want several valid results: each should reflect the reference smoke and maintain the original scene composition, while the smoke plume can vary in shape, density, and precise placement. The paired outputs in Figure 7 illustrate variations from noise samples. During fine-tuning, however, shared objectives can draw all N fixed-noise candidates toward similar appearances. KINESYS includes a diversity term $\mathcal{D}$ to discourage early convergence among the fixed-noise candidates.

Following repulsive particle objectives (Liu & Wang, 2019; Corso et al., 2023), $\mathcal{D}$ averages radial basis function (RBF) similarities across distinct candidates. The complete optimization objective combines the weighted candidate losses with this diversity penalty:

$$\mathcal{L}_{\text{pop}} = \sum_{i=1}^{N} \alpha_i \mathcal{L}_i + \lambda_{\text{div}}(t)\mathcal{D} \qquad \text{(3a)} \qquad \mathcal{D} = \frac{1}{N(N-1)} \sum_{i=1}^{N} \sum_{j \neq i} \exp\left(-\frac{d_{ij}}{h_i}\right) \qquad \text{(3b)}$$

Here, d_{ij} is the mean squared distance within the editable region between candidate i's current latent and candidate j's latent from the previous optimization step. The bandwidth h_i is based on the median distance to the other candidates. We hold the previous-step latents and bandwidth fixed during gradient computation, so minimizing $\mathcal{D}$ pushes each current latent away from the other candidates' recent states. For $N = 1$, we set $\mathcal{D} = 0$. The gradient $\nabla_\phi \mathcal{L}_{\text{pop}}$ updates only the LoRA parameters ϕ. The weight $\lambda_{\text{div}}(t)$ is positive during early updates to discourage candidate collapse, then decays to zero so later updates focus on reference-grounded VQA correctness and EditReward's instruction-following and visual-quality signals.

Independent fine-tuning runs. Candidate weighting reduces the influence of poor edits, but all N candidates still shape one LoRA. Putting all samples into one run would keep their updates coupled, so a promising starting set could still be pulled toward a compromise with less useful ones. KINESYS instead starts $K = 5$ independent runs, each with its own LoRA parameters ϕ_k and its own N fixed-noise samples. These K sets of N noise draws allow different starting sets to improve independently; the runs exchange neither parameters nor candidates. The candidate and run study in Section 4.3 supports this compute-limited configuration: three candidates score highest among the tested widths, and multiple independent runs improve the lower-of-two score over one run.

Pruning and checkpoint selection. Not every starting set deserves the full tuning budget. KINESYS therefore saves intermediate LoRA checkpoints and uses the first two scheduled checkpoint boundaries to rank runs by recent VQA and EditReward feedback. Lower-ranked runs stop, leaving three after the first pruning point and two after the second. The counts limit compute, but the search remains open across multiple runs until a separate step selects one saved checkpoint.

To favor a checkpoint that works beyond the fixed noise samples used for fine-tuning, KINESYS renders an evaluation image using the same held-out noise sample at every saved checkpoint. It

selects the LoRA parameters $\phi^\star$ whose evaluation image has the highest EditReward score. The selected parameters give the artist an adapted editor that can produce further effect variations from fresh noise while retaining the source scene. The Supplementary Material includes implementation details, prompt templates, and additional visual results.

4 EXPERIMENTS

We test whether KINESYS transfers a reference effect while preserving source identity. We compare it with open-weight and proprietary editors in Sec. 4.2 and perform ablation studies in Sec. 4.3.

4.1 EXPERIMENTAL SETUP

Dataset and tasks. Our evaluation contains 81 effect-reference editing requests across eight categories: boat wakes, vehicle dents, explosions, fire, puddles, cast shadows, smoke, and snow. The requests use 78 distinct source photographs and 63 effect references. Each edit request includes a short effect-editing prompt and three yes/no questions reused within its effect category. These questions test placement, realism, and a defining effect property against the reference image. The supplementary material provides more details on dataset construction.

Baselines. We compare KINESYS with six direct open-weight editors: FLUX.2 Klein 4B (Black Forest Labs, 2026), Qwen-Image-Edit (Qwen Team, 2025), OmniGen2 (Wu et al., 2026a), Step1X-Edit v1.2 (Liu et al., 2025a), ACE++ (Mao et al., 2025a), and ICEdit (Zhang et al., 2025), plus proprietary Nano Banana 2 (Gemini 3.1 Flash Image) (Raisinghani, 2026). These direct editors use the editing instruction and generate a result without per-request tuning or the reference questions.

Native FLUX.2 Klein 4B receives the same source, effect reference, mask, and generation settings as KINESYS, serving as its unadapted control. Our adapted Dual-Process baseline (Luo et al., 2025) fine-tunes the same image-conditioned generator with the same instruction, source, reference, and three questions as KINESYS, using only VQA feedback. For the automated comparison, KINESYS uses a full-image mask because some baselines do not support spatial masks.

KINESYS configuration. For the main comparison and feedback ablation, a frozen Gemma 4 E4B-it VLM (Google DeepMind, 2026) answers the reference-grounded questions. Each run uses $N = 3$ fixed-noise candidates for up to $T = 100$ steps, saving checkpoints every 15 steps. KINESYS starts $K = 5$ independent fine-tuning runs and uses VQA and EditReward feedback to retain three at the first pruning point and two at the second.

Evaluation protocol. We evaluate each of the 81 requests under two seeds to reduce dependence on a single random draw. For each seed, KINESYS starts a separate set of fine-tuning runs and selects a checkpoint independently. We compare methods on the 156 of 162 request–seed cases completed by every method. Six cases lack outputs, mainly due to Nano Banana 2's policy blocks.

We prompt a held-out Gemini 2.5 Flash judge (Comanici et al., 2025) with the source image, effect reference, edited image, and instruction. The judge reports 0–10 scores for perceptual quality (realism and artifacts) and source consistency (preservation of untargeted content). Table 1 reports the average of these two scores. EditScore-72B (Luo et al., 2026) and EditReward (Wu et al., 2026b) are learned judges that receive the source, instruction, and edited image; neither sees the effect reference. EditScore-72B combines semantic consistency (instruction following and preservation of unrequested content) with perceptual quality (realism and artifacts) into an overall edit-quality score. EditReward is trained on human preferences and predicts separate instruction-following and visual-quality scores, which we average.

We compare each edit with its source using LPIPS distance (Zhang et al., 2018) and TPIPS under its "overall" setting (Wang et al., 2026). Gemini and EditScore-72B judge broader editing quality that includes source consistency; LPIPS and TPIPS are source-similarity proxies for identity preservation. An unchanged edit can score well on similarity while failing to transfer the effect, so we interpret these measures alongside the visual results and user study.

User study. Because automated judges are imperfect proxies for human judgment, we also ran a blinded user study; the final analysis includes 16 edits, two per effect category. We compared KINESYS with Step1X-Edit v1.2, the highest-scoring evaluated open-weight editor in Table 1, and with

Table 1: Comparison on 156 shared request–seed cases (81 requests). Open-weight methods are ordered by Gemini; bold and underline mark first and second. †: EditReward used for tuning and selection (non-independent). ‡: generator–judge family overlap.

	Automated judges			Identity preservation	
Method	Gemini↑	Edit Score↑	Edit Reward↑	LPIPS↓	TPIPS↑
Open-weight methods					
OmniGen2 (2026a)	4.95	7.53	−0.018	0.753	0.599
ACE++ (2025a)	4.98	5.95	−0.503	0.795	0.500
ICEdit (2025)	5.93	6.91	−0.097	<u>0.429</u>	0.632
Dual-Process (2025)	7.09	7.93	0.212	0.577	0.615
Qwen-Image-Edit (2025)	7.67	8.31	0.435	0.645	0.599
FLUX.2 Klein 4B (2026)	7.90	8.68	0.501	0.579	0.640
Step1X-Edit (2025a)	8.22	8.80	0.666	0.449	<u>0.724</u>
Proprietary methods					
Nano Banana 2 (2026)	**8.91**‡	**9.06**	<u>0.738</u>	<u>0.429</u>	0.696
KINESYS (Ours)	<u>8.83</u>	<u>8.95</u>	**1.007**†	**0.377**	**0.731**

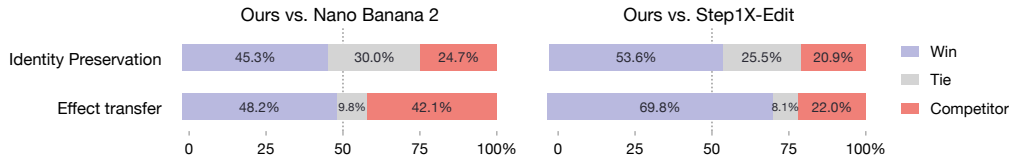

Figure 3: Blinded paired-comparison user study on 16 edits, with 41 responses for Nano Banana 2 and 40 for Step1X-Edit. Bars show preferences for identity preservation and effect transfer; ties indicate users voting for no clear winner.

proprietary Nano Banana 2. We report 41 responses for Nano Banana 2 and 40 for Step1X-Edit; most respondents are not VFX experts. For each baseline, the retained analysis covers 16 comparisons per response, judging identity preservation and effect transfer separately, with ties allowed. Figure 3 reports preferences. The supplement includes an example survey question for reference.

4.2 Comparison with Existing Editors

KINESYS achieves the strongest source-preservation scores among the evaluated editors: LPIPS distance is 0.377 versus Nano Banana 2's 0.429, and TPIPS similarity is 0.731 versus 0.696. It also leads the evaluated open-weight editors on Gemini and EditScore-72B. The user study in Figure 3 reinforces the preservation result. KINESYS was preferred for identity preservation over Nano Banana 2 (45% versus 25%) and Step1X-Edit (54% versus 21%). For effect transfer, preferences were close against Nano Banana 2 (48% versus 42%) and favored KINESYS over Step1X-Edit (70% versus 22%). Figure 1 also shows scene changes in the 'puddle' edit and an extra boat in Nano Banana 2's 'boat-wake' edit demonstrating the gap in identity preservation ability.

4.3 Ablation

Effect transfer and identity preservation. We highlight the need for two complementary VLM judges in Figure 5. Without the EditReward gradient, the effect appears but unrelated source content changes: a building appears in the explosion edit, and the boat changes appearance and position in the wake edit. Without the VQA gradient, the scene stays closer to the source, but the explosion loses the reference's form, and the wake misses its intended pattern. With both gradients, KINESYS better balances effect transfer and identity preservation in these examples.

Candidate and fine-tuning run configurations. The numbers of candidates and independent runs affect both editing quality and per-request compute. We use $N = 3$ candidates and $K = 5$ runs, and vary each setting separately in Table 2 to assess that balance. Increasing N from 1 to 3 raises mean

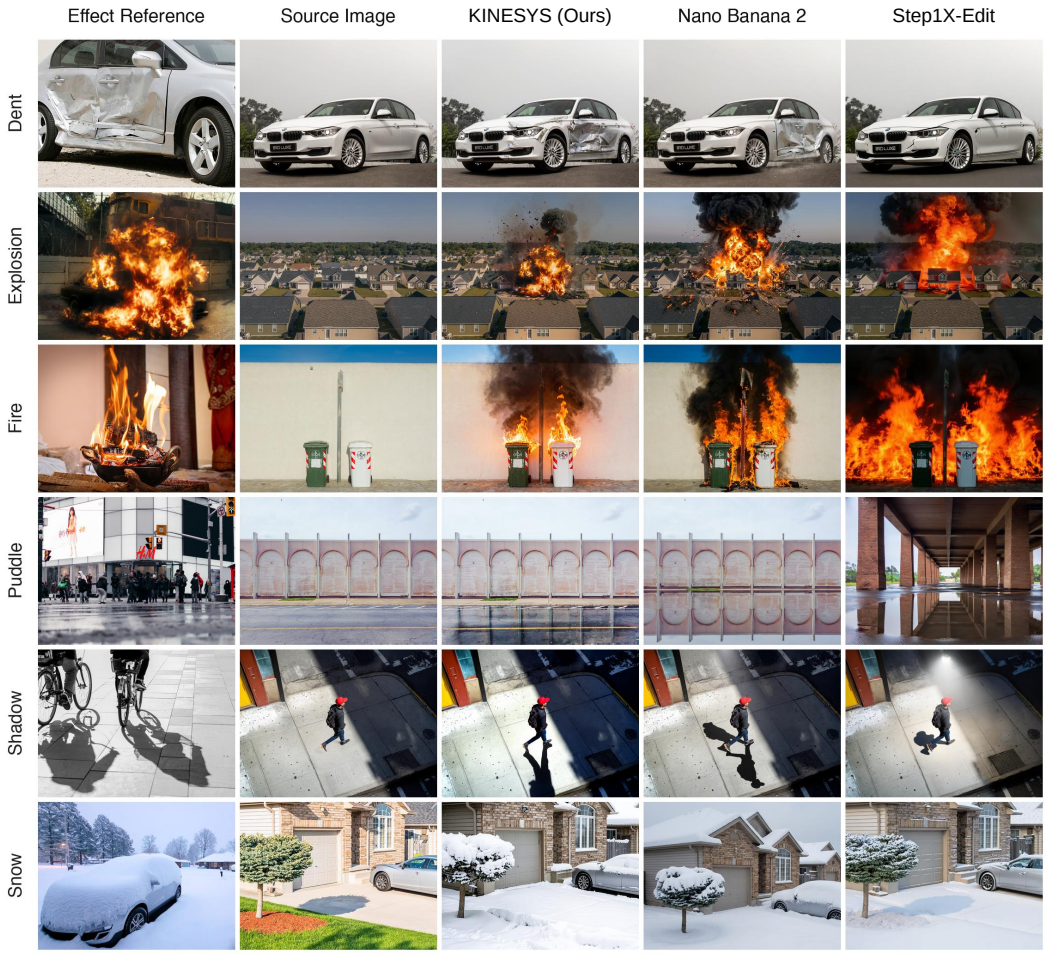

Figure 4: Six-effect comparison. Columns: reference, source, KINESYS, Nano Banana 2, Step1X-Edit v1.2. In the snow row, KINESYS retains the source house and camera perspective while Nano Banana 2 changes both. In the fire row, KINESYS adds flames to the trash cans without the scene-wide blaze of Step1X-Edit.



Figure 5: Feedback ablations on explosion and wake edits. The "w/o" labels denote removal of the named gradient; red boxes mark identity changes without EditReward. Without VQA, effect form departs from the reference. Full KINESYS preserves source content and transfers the effect.

Gemini from 8.59 to 9.00 and the mean worst-seed score from 8.11 to 8.73. At $N = 5$, these fall to 8.55 and 7.80. One possible explanation is that feedback from a wider set of fixed-noise candidates pulls their shared adaptation toward a less useful compromise.

Table 2: Gemini scores for 16 requests, varying candidate count N and independent runs K. Mean averages both seeds; Worst averages the lower seed score per request.

(a) Candidate edits per run

N	Mean↑	Worst↑
1	8.59	8.11
3 **(ours)**	9.00	8.73
5	8.55	7.80

(b) Independent fine-tuning runs

K	Mean↑	Worst↑
1	7.31	6.45
3	9.02	8.59
5 **(ours)**	9.00	8.73
8	9.12	8.88

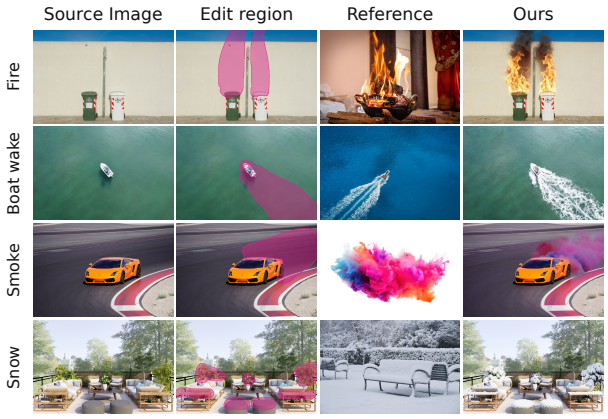

Figure 6: Spatially controlled effect-reference edits. Columns show the source, pink edit region, effect reference, and KINESYS result.

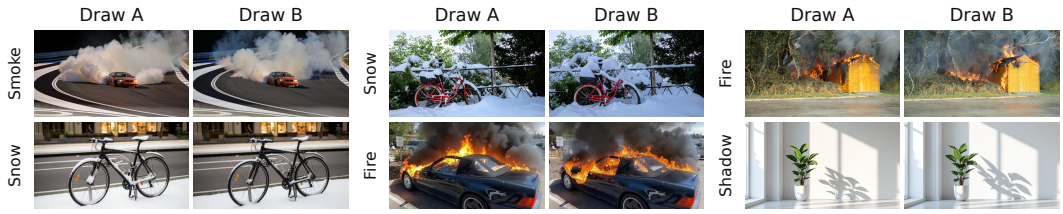

Figure 7: Fresh-noise variations in smoke shape, snow intensity, fire extent, and cast shadows. Each pair uses the same source, reference, instruction, and selected LoRA; only the noise changes.

Increasing K from one to five runs raises mean Gemini from 7.31 to 9.00 and the mean worst-seed score from 6.45 to 8.73. Three runs have a slightly higher mean (9.02) but a lower worst-seed score (8.59); eight runs improve both to 9.12 and 8.88 at greater compute. We use five runs to favor worst-seed quality without the cost of eight.

4.4 APPLICATIONS

Visual-effects artists may know the desired effect and where it belongs, yet want several convincing versions before choosing one (Figure 1, top). KINESYS offers both forms of control: a mask localizes the effect while the surrounding scene remains consistent (Figure 6), and sampling the selected checkpoint with fresh noise changes details such as snow amount, smoke shape and density, or burn area (Figure 7). The requested effect and source scene remain consistent across those draws. Editing quality holds across five unseen draws from each of 162 selected checkpoints: mean EditScore-72B is 8.91, compared with 8.93 for the selected outputs.

5 CONCLUSION

KINESYS adapts a generator to transfer a reference-demonstrated effect while preserving the source shot's identity. Complementary VLM feedback and independent fine-tuning runs support this balance, while fresh-noise samples offer multiple edit variations. Across 81 requests, KINESYS outperforms Nano Banana 2 on identity preservation as measured by LPIPS, TPIPS, and blind-study preference, while remaining competitive on editing quality and effect transfer. It leads the evaluated open-weight editors on Gemini and EditScore-72B.

The approach still depends on what its generator can render and its VLM judges can recognize: an unfamiliar creature with a rare stylized electrical effect, for example, may be difficult to edit faithfully. Bad starting noise can still yield a weak edit, and per-request fine-tuning adds latency. KINESYS is a first step towards frameworks that enable effect transfer without losing source identity.

6 AI USE STATEMENT

Generative AI tools, including OpenAI Codex and Claude Code, were used to assist with manuscript organization and editing, literature review, figure planning, code implementation and review, and discussion of experimental design and result interpretation. We find that AI tools are useful as another set of eyes to help give a fresh reviewer-like perspective on ideas that we as the authors may have bias on after working with them for so long. Naturally AI is strong at coding so quickly taking our ideas from discussion to implementation has been a big part of how we have used it in the research process. We have reviewed all AI-assisted work and verified the reported methods, numerical results, citations, figures, and claims against the underlying source code and experimental artifacts. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Chengyu Bai, Jintao Chen, Xiang Bai, Yilong Chen, Qi She, Ming Lu, and Shanghang Zhang. Uniedit-i: Training-free image editing for unified vlm via iterative understanding, editing and verifying, 2025. URL https://arxiv.org/abs/2508.03142.

Heli Ben-Hamu, Omri Puny, Itai Gat, Brian Karrer, Uriel Singer, and Yaron Lipman. D-flow: Differentiating through flows for controlled generation, 2024. URL https://arxiv.org/abs/2402.14017.

Elior Benarous, Yilun Du, and Heng Yang. Spie: Semantic and structural post-training of image editing diffusion models with ai feedback, 2025. URL https://arxiv.org/abs/2504.12833.

Black Forest Labs. FLUX.2: Official inference repository. GitHub repository, 2026. URL https://github.com/black-forest-labs/flux2.

Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions, 2023. URL https://arxiv.org/abs/2211.09800.

Jinho Chang, Jaemin Kim, and Jong Chul Ye. Training-free reward-guided image editing via trajectory optimal control, 2026. URL https://arxiv.org/abs/2509.25845.

Sherry X. Chen, Yi Wei, Luowei Zhou, and Suren Kumar. Adiee: Automatic dataset creation and scorer for instruction-guided image editing evaluation, 2025. URL https://arxiv.org/abs/2507.07317.

Xi Chen, Yutong Feng, Mengting Chen, Yiyang Wang, Shilong Zhang, Yu Liu, Yujun Shen, and Hengshuang Zhao. Zero-shot image editing with reference imitation, 2024. URL https://arxiv.org/abs/2406.07547.

Kevin Clark, Paul Vicol, Kevin Swersky, and David J Fleet. Directly fine-tuning diffusion models on differentiable rewards, 2024. URL https://arxiv.org/abs/2309.17400.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities, 2025. URL https://arxiv.org/abs/2507.06261.

Gabriele Corso, Yilun Xu, Valentin de Bortoli, Regina Barzilay, and Tommi Jaakkola. Particle guidance: non-i.i.d. diverse sampling with diffusion models, 2023. URL https://arxiv.org/abs/2310.13102.

Haiwen Diao, Penghao Wu, Hanming Deng, Jiahao Wang, Shihao Bai, Silei Wu, Weichen Fan, Wenjie Ye, Wenwen Tong, Xiangyu Fan, Yan Li, Yubo Wang, Zhijie Cao, Zhiqian Lin, Zhitao Yang, Zhongang Cai, Yuwei Niu, Yue Zhu, Bo Liu, Chengguang Lv, Haojia Yu, Haozhe Xie, Hongli Wang, Jianan Fan, Jiaqi Li, Jiefan Lu, Jingcheng Ni, Junxiang Xu, Kaihuan Liang, Lianqiang Shi, Linjun Dai, Linyan Wang, Oscar Qian, Peng Gao, Pengfei Liu, Qingping Sun, Rui Shen, Ruisi Wang, Shengnan Ma, Shuang Yang, Siyi Xie, Siying Li, Tianbo Zhong, Xiangli Kong, Xuanke Shi, Yang Gao, Yongqiang Yao, Yves Wang, Zhengqi Bai, Zhengyu Lin, Zixin Yin, Wenxiu

Sun, Ruihao Gong, Quan Wang, Lewei Lu, Lei Yang, Ziwei Liu, and Dahua Lin. Sensenova-u1: Unifying multimodal understanding and generation with neo-unify architecture, 2026. URL https://arxiv.org/abs/2605.12500.

Google DeepMind. Gemma 4. Model card, 2026. URL https://deepmind.google/models/gemma/gemma-4/.

Rogerio Guimaraes and Pietro Perona. Inference-time scaling of diffusion models via progressive seed pruning, 2026. URL https://arxiv.org/abs/2607.21591.

Jiaqi Guo, Lianli Gao, Junchen Zhu, Jiaxin Zhang, Siyang Li, and Jingkuan Song. MagicVFX: Visual effects synthesis in just minutes. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 8238–8246, 2024. doi: 10.1145/3664647.3681516.

Runze He, Kai Ma, Linjiang Huang, Shaofei Huang, Jialin Gao, Xiaoming Wei, Jiao Dai, Jizhong Han, and Si Liu. Freeedit: Mask-free reference-based image editing with multi-modal instruction, 2024. URL https://arxiv.org/abs/2409.18071.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. URL https://arxiv.org/abs/2106.09685.

Zeyu Jiang, Lai Man Po, Xuyuan Xu, Yexin Wang, Guoping Gong, Haoxuan Wu, Chenbo Yan, Kun Li, and Yuyang Liu. OrionEdit: Bridging reference and source images for generalized cross-image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9127–9138, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Jiang_OrionEdit_Bridging_Reference_and_Source_Images_for_Generalized_Cross-Image_Editing_CVPR_2026_paper.html.

Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models, 2023. URL https://arxiv.org/abs/2210.09276.

Jaihoon Kim, Taehoon Yoon, Jisung Hwang, and Minhyuk Sung. Inference-time scaling for flow models via stochastic generation and rollover budget forcing, 2025. URL https://arxiv.org/abs/2503.19385.

Nupur Kumari, Sheng-Yu Wang, Nanxuan Zhao, Yotam Nitzan, Yuheng Li, Krishna Kumar Singh, Richard Zhang, Eli Shechtman, Jun-Yan Zhu, and Xun Huang. Learning an image editing model without image editing pairs, 2026. URL https://arxiv.org/abs/2510.14978.

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. URL https://arxiv.org/abs/2506.15742.

Tiancheng Li, Jinxiu Liu, Huajun Chen, and Qi Liu. Instructrl4pix: Training diffusion for image editing by reinforcement learning, 2024. URL https://arxiv.org/abs/2406.09973.

Yan Li, Lin Liu, Xiaopeng Zhang, Wei Xue, Wenhan Luo, Yike Guo, and Qi Tian. Cogniedit: Dense gradient flow optimization for fine-grained image editing, 2025. URL https://arxiv.org/abs/2512.13276.

Qiang Liu and Dilin Wang. Stein variational gradient descent: A general purpose bayesian inference algorithm, 2019. URL https://arxiv.org/abs/1608.04471.

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, 2025a. URL https://arxiv.org/abs/2504.17761.

Xinyu Liu, Ailing Zeng, Wei Xue, Harry Yang, Wenhan Luo, Qifeng Liu, and Yike Guo. Vfx creator: Animated visual effect generation with controllable diffusion transformer, 2025b. URL https://arxiv.org/abs/2502.05979.

Grace Luo, Jonathan Granskog, Aleksander Holynski, and Trevor Darrell. Dual-process image generation, 2025. URL https://arxiv.org/abs/2506.01955.

Xin Luo, Jiahao Wang, Chenyuan Wu, Shitao Xiao, Xiyan Jiang, Defu Lian, Jiajun Zhang, Dong Liu, and Zheng liu. Editscore: Unlocking online rl for image editing via high-fidelity reward modeling, 2026. URL https://arxiv.org/abs/2509.23909.

Nanye Ma, Shangyuan Tong, Haolin Jia, Hexiang Hu, Yu-Chuan Su, Mingda Zhang, Xuan Yang, Yandong Li, Tommi Jaakkola, Xuhui Jia, and Saining Xie. Inference-time scaling for diffusion models beyond scaling denoising steps, 2025. URL https://arxiv.org/abs/2501.09732.

Chaojie Mao, Jingfeng Zhang, Yulin Pan, Zeyinzi Jiang, Zhen Han, Yu Liu, and Jingren Zhou. Ace++: Instruction-based image creation and editing via context-aware content filling, 2025a. URL https://arxiv.org/abs/2501.02487.

Fangyuan Mao, Aiming Hao, Jintao Chen, Dongxia Liu, Xiaokun Feng, Jiashu Zhu, Meiqi Wu, Chubin Chen, Jiahong Wu, and Xiangxiang Chu. Omni-effects: Unified and spatially-controllable visual effects generation, 2025b. URL https://arxiv.org/abs/2508.07981.

Yingmao Miao, Pengfei Zhang, Xiaochen Lv, Meng Yu, Lei Sun, Xiangxiang Chu, Chao Shen, and Chenhao Lin. Evaluation-verification reward for consistent multi-reference image editing, 2026. URL https://arxiv.org/abs/2607.29025.

Thao Nguyen, Yuheng Li, Utkarsh Ojha, and Yong Jae Lee. Visual instruction inversion: Image editing via visual prompting, 2023. URL https://arxiv.org/abs/2307.14331.

OpenAI. GPT-Image-2 model. OpenAI API Documentation, 2026. URL https://developers.openai.com/api/docs/models/gpt-image-2.

Yulin Pan, Xiangteng He, Chaojie Mao, Zhen Han, Zeyinzi Jiang, Jingfeng Zhang, and Yu Liu. Ice-bench: A unified and comprehensive benchmark for image creating and editing, 2025. URL https://arxiv.org/abs/2503.14482.

Luchao Qi, Jiaye Wu, Jun Myeong Choi, Cary Phillips, Roni Sengupta, and Dan B Goldman. Over++: Generative video compositing for layer interaction effects, 2025. URL https://arxiv.org/abs/2512.19661.

Xiangyan Qu, Zhenlong Yuan, Jing Tang, Rui Chen, Datao Tang, Meng Yu, Lei Sun, Yancheng Bai, Xiangxiang Chu, Gaopeng Gou, Gang Xiong, and Yujun Cai. From scale to speed: Adaptive test-time scaling for image editing, 2026. URL https://arxiv.org/abs/2603.00141.

Qwen Team. Qwen-Image-Edit: Image editing with higher quality and efficiency. Qwen Blog, 2025. URL https://qwenlm.github.io/blog/qwen-image-edit/.

Naina Raisinghani. Nano Banana 2: Combining pro capabilities with lightning-fast speed. Google, 2026. URL https://blog.google/innovation-and-ai/technology/ai/nano-banana-2/.

Amirmojtaba Sabour, Michael S. Albergo, Carles Domingo-Enrich, Nicholas M. Boffi, Sanja Fidler, Karsten Kreis, and Eric Vanden-Eijnden. Test-time scaling of diffusions with flow maps, 2025. URL https://arxiv.org/abs/2511.22688.

Onkar Susladkar, Dong-Hwan Jang, Tushar Prakash, Adheesh Juvekar, Vedant Shah, Ayush Barik, Nabeel Bashir, Muntasir Wahed, Ritish Shrirao, and Ismini Lourentzou. Rewardflow: Generate images by optimizing what you reward, 2026. URL https://arxiv.org/abs/2604.08536.

Team Seedream, Yunpeng Chen, Yu Gao, Lixue Gong, Meng Guo, Qiushan Guo, Zhiyao Guo, Xiaoxia Hou, Weilin Huang, Yixuan Huang, Xiaowen Jian, Huafeng Kuang, Zhichao Lai, Fanshi Li, Liang Li, Xiaochen Lian, Chao Liao, Liyang Liu, Wei Liu, Yanzuo Lu, Zhengxiong Luo, Tongtong Ou, Guang Shi, Yichun Shi, Shiqi Sun, Yu Tian, Zhi Tian, Peng Wang, Rui Wang, Xun Wang, Ye Wang, Guofeng Wu, Jie Wu, Wenxu Wu, Yonghui Wu, Xin Xia, Xuefeng Xiao, Shuang Xu, Xin Yan, Ceyuan Yang, Jianchao Yang, Zhonghua Zhai, Chenlin Zhang, Heng Zhang, Qi Zhang, Xinyu Zhang, Yuwei Zhang, Shijia Zhao, Wenliang Zhao, and Wenjia Zhu. Seedream 4.0: Toward next-generation multimodal image generation, 2025. URL https://arxiv.org/abs/2509.20427.

Dani Valevski, Matan Kalman, Eyal Molad, Eyal Segalis, Yossi Matias, and Yaniv Leviathan. Unitune: Text-driven image editing by fine tuning a diffusion model on a single image, 2023. URL https://arxiv.org/abs/2210.09477.

Qian Wang, Aleksandar Cvejic, Abdelrahman Eldesokey, and Peter Wonka. Editclip: Representation learning for image editing, 2025. URL https://arxiv.org/abs/2503.20318.

Sheng-Yu Wang, Yotam Nitzan, Aaron Hertzmann, Jun-Yan Zhu, Eli Shechtman, Alexei A. Efros, and Richard Zhang. The many senses of visual similarity: A text-prompted image perceptual metric, 2026. URL https://arxiv.org/abs/2607.18237.

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. URL https://arxiv.org/abs/2508.02324.

Chenyuan Wu, Pengfei Zheng, Ruiran Yan, Shitao Xiao, Xin Luo, Yueze Wang, Wanli Li, Xiyan Jiang, Yexin Liu, Junjie Zhou, Ze Liu, Ziyi Xia, Chaofan Li, Haoge Deng, Jiahao Wang, Kun Luo, Bo Zhang, Defu Lian, Xinlong Wang, Zhongyuan Wang, Tiejun Huang, and Zheng Liu. Omnigen2: Towards instruction-aligned multimodal generation, 2026a. URL https://arxiv.org/abs/2506.18871.

Keming Wu, Sicong Jiang, Max Ku, Ping Nie, Minghao Liu, and Wenhu Chen. Editreward: A human-aligned reward model for instruction-guided image editing, 2026b. URL https://arxiv.org/abs/2509.26346.

Binxin Yang, Shuyang Gu, Bo Zhang, Ting Zhang, Xuejin Chen, Xiaoyan Sun, Dong Chen, and Fang Wen. Paint by example: Exemplar-based image editing with diffusion models, 2022. URL https://arxiv.org/abs/2211.13227.

Shiyuan Yang, Ruihuang Li, Jiale Tao, Shuai Shao, Qinglin Lu, and Jing Liao. Effectmaker: Unifying reasoning and generation for customized visual effect creation, 2026. URL https://arxiv.org/abs/2603.06014.

Yue Yu, Yang Jiao, Jiayu Wang, Qi Dai, and Jingjing Chen. Adaptive inference-time scaling via early-step latent verification for image editing, 2026. URL https://arxiv.org/abs/2606.15188.

Ziyun Zeng, Hang Hua, and Jiebo Luo. Mira: Multimodal iterative reasoning agent for image editing, 2026. URL https://arxiv.org/abs/2511.21087.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric, 2018. URL https://arxiv.org/abs/1801.03924.

Shu Zhang, Xinyi Yang, Yihao Feng, Can Qin, Chia-Chih Chen, Ning Yu, Zeyuan Chen, Huan Wang, Silvio Savarese, Stefano Ermon, Caiming Xiong, and Ran Xu. Hive: Harnessing human feedback for instructional visual editing, 2024. URL https://arxiv.org/abs/2303.09618.

Zechuan Zhang, Ji Xie, Yu Lu, Zongxin Yang, and Yi Yang. In-context edit: Enabling instructional image editing with in-context generation in large scale diffusion transformer, 2025. URL https://arxiv.org/abs/2504.20690.

Baixuan Zhao, Xinyu Zhang, Huayu Zheng, Shuaicheng Liu, Xiongkuo Min, Guangtao Zhai, and Xiaohong Liu. From open loop to closed loop: A test-time iterative optimization framework for reference-consistent image generation, 2026. URL https://arxiv.org/abs/2607.04691.

Rui Zhao and Mike Zheng Shou. P-flow: Prompting visual effects generation, 2026. URL https://arxiv.org/abs/2603.22091.