PASEDIT: PORTRAIT AESTHETIC STRUCTURAL EDITING

Anonymous authors
Paper under double-blind review

ABSTRACT

Generative image editing models enable high-level structural editing of portraits for aesthetic enhancement, such as adjusting pose, viewpoint, and framing. However, existing editors still face two key challenges: deciding how to improve a structurally suboptimal portrait from a coarse request, and making effective structural changes without causing identity drift or unintended changes. We propose **PASEdit**, a planner-executor framework for controllable single-factor portrait structural editing across pose, viewpoint, and framing. For planning, we introduce **Aesthetic Planning Post-Training (AesPT)**, which trains factor-specific experts with task-specific visual feedback and consolidates their capabilities into a unified planner through multi-teacher on-policy distillation. For execution, the executor first learns structural transformations from paired supervision and is then optimized for identity and non-target content preservation while maintaining instruction adherence. To provide controlled supervision for structural learning, we construct 12,422 portrait editing pairs derived from real photographic degradation patterns. Extensive experiments show that PASEdit achieves more effective structural improvements while maintaining strong identity and non-target content preservation.

1 INTRODUCTION

Traditional photographic post-processing improves images mainly through adjustments to exposure, color, tone, and local appearance (Liang et al., 2021). Generative image editing models further enable high-level structural editing of portraits for aesthetic enhancement, such as adjusting human pose, camera viewpoint, and framing after capture (Wu et al., 2025; Du et al., 2026). These factors directly affect how people are presented and the visual expressiveness of an image.

A coarse request for photographic aesthetic improvement may leave the desired structural changes unspecified. The model must then determine how to adjust the image before carrying out the edit. For portraits, this decision must also satisfy strict identity preservation constraints. AesFormer (Du et al., 2026) separates aesthetic planning from image editing for holistic multi-factor reconstruction. However, the broad editing freedom of multi-factor reconstruction can make it difficult to preserve subject identity and background content. We therefore propose **PASEdit**, a planner-executor framework that targets one structural factor at a time (pose, viewpoint, or framing) while seeking to preserve subject identity and non-target content. The planner turns a coarse request into a concrete single-factor editing instruction, while the executor applies it to the image. During supervised fine-tuning, we include open-ended requests paired with single-factor editing instructions. Figure 1 shows results for both requests that specify a structural factor and open-ended improvement requests.

Planning faces two challenges. First, a single portrait often admits multiple reasonable structural improvement plans. Second, pose, viewpoint, and framing rely on different visual criteria, which a single planning strategy may struggle to fully capture. We therefore introduce **Aesthetic Planning Post-Training (AesPT)**, a task-specialized post-training strategy for the planner. Starting from a shared supervised initialization, AesPT develops factor-specific planning capabilities through reinforcement learning with structured feedback designed for each structural factor, and then consolidates them within a unified planner through multi-teacher on-policy distillation (Agarwal et al., 2024; Ma et al., 2026).

Execution must balance structural changes with content preservation. The executor first learns structural transformations from paired supervision and is then further optimized through preservation-aware post-training for identity and non-target content preservation while maintaining

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.

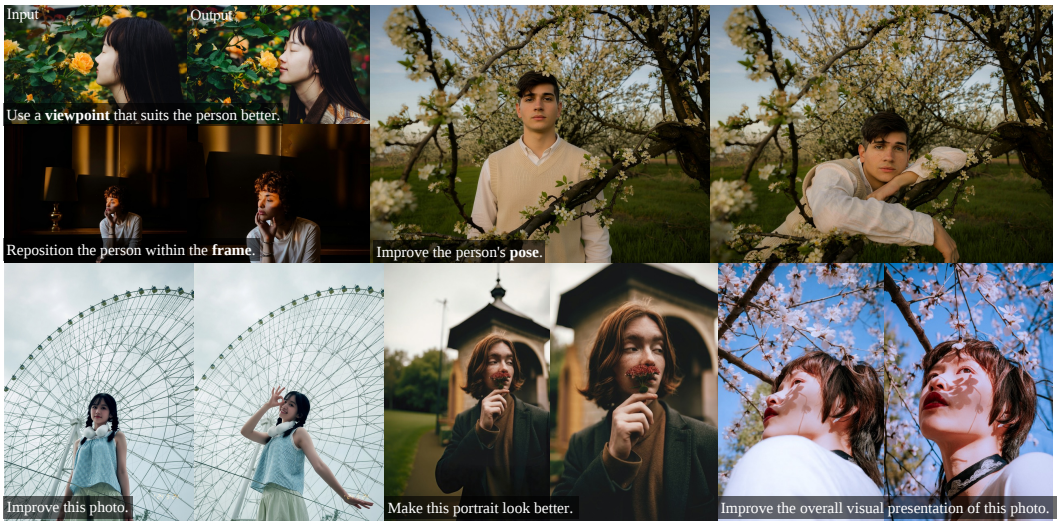


Figure 1: **Portrait aesthetic structural editing.** PASEdit turns coarse requests into one pose, viewpoint, or framing edit while seeking to preserve identity and non-target content. Upper examples specify the factor; lower examples use open-ended requests.

instruction adherence. This allows it to make effective structural changes while reducing identity drift and unintended changes to the remaining image content.

To provide controlled supervision for structural learning, we derive reusable structural degradation patterns from real photographic examples and apply them to diverse, high-quality portraits. We then validate the resulting pairs to construct 12,422 controlled portrait structural editing pairs covering pose, viewpoint, and framing.

Experiments show that PASEdit achieves more effective structural improvements while preserving subject identity and non-target content.

Our main contributions are:

- We develop PASEdit, a framework for controllable single-factor portrait structural editing across pose, viewpoint, and framing, while seeking to preserve identity and non-target content.
- We introduce AesPT, a task-specialized post-training strategy that develops factor-specific planning capabilities and consolidates them within a unified planner.
- We derive structural degradation patterns from real photographic examples and use them to construct 12,422 controlled portrait structural editing pairs for training and evaluation.

2 Related Work

Portrait aesthetics and structural editing. Existing portrait enhancement methods mainly focus on appearance-level aesthetics, including portrait retouching and facial beautification (Liang et al., 2021; Xie et al., 2023; Xu et al., 2025; Yang et al., 2026). Recent work has further explored higher-level aesthetic understanding and structural manipulation. AesExpert studies multimodal aesthetic perception (Huang et al., 2024). PhotoFramer studies composition guidance for general photographs through shift, zoom-in, and view-change operations, generating textual guidance together with illustrative examples (You et al., 2026). PortraitCraft investigates fine-grained portrait composition understanding and generation (Sha et al., 2026), while PhotoAgent explores multi-step aesthetic planning with visual feedback (Yao et al., 2026). Most closely related, AesFormer separates aesthetic planning from structural editing for holistic multi-factor reconstruction (Du et al., 2026). Our work studies controllable single-factor portrait structural editing over pose, viewpoint, and framing.

Reward optimization and capability consolidation. GRPO optimizes language-model policies using group-relative advantages without a learned value model (Shao et al., 2024). On-policy distillation trains a student on its own generated prefixes (Agarwal et al., 2024), while MOPD extends this paradigm to consolidate multiple domain-specialized RL teachers into a unified model (Ma et al., 2026). For image generative models, DDPO introduces reinforcement learning for diffusion models (Black et al., 2024), while Flow-GRPO and DiffusionNFT develop reward-based post-training based

on flow matching (Liu et al., 2025; Zheng et al., 2026). **These methods provide the optimization foundations for the planner and executor post-training used in our framework.**

3 Controlled Portrait Pairs

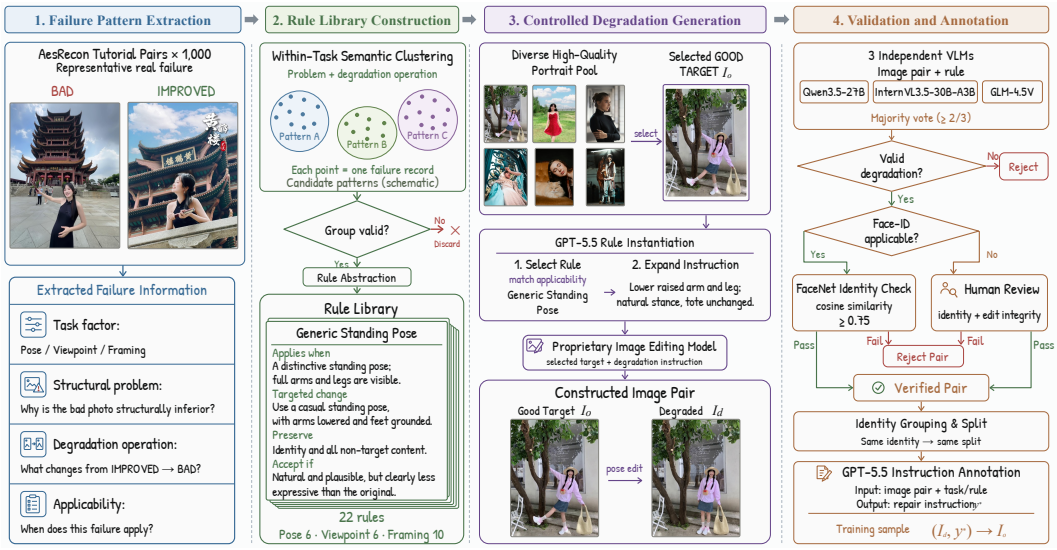


Figure 2: **Construction pipeline for controlled portrait structural editing pairs.** We derive reusable degradation rules from real photographic examples and apply them to high-quality portraits to generate degraded inputs I_{d}.

Portrait structural editing requires the improvement in each training pair to be clearly attributable to a single target structural factor. Randomly generated structural degradations are easy to scale, but may introduce artificial failures that depart from real photographic distributions; real before-and-after photographs often change multiple factors simultaneously, making them unsuitable for direct use as strict supervision. We therefore derive degradation rules from real photographic examples, apply them to diverse high-quality portraits, and construct controlled editing pairs through multi-stage verification. Figure 2 presents the overall pipeline.

Failure patterns and rule library. We use GPT-5.5 (OpenAI, 2026a) to analyze 1,000 poor–improved photography tutorial pairs from AesRecon (Du et al., 2026). For each example, we extract the structural task, the specific structural problem, the degradation operation from improved to poor, and its applicability conditions. We then semantically cluster these failure records within each task and verify whether each candidate cluster represents a consistent failure pattern. Valid clusters are abstracted into reusable rules specifying applicability conditions, degradation operations, and content to preserve. This yields 22 rules: six for pose, six for viewpoint, and ten for framing.

Controlled degradation. To expand the high-quality target image pool and its diversity in subjects, scenes, and compositions, we supplement AesRecon images with 5,000 high-quality Unsplash portraits from PortraitCraft (Sha et al., 2026; Lou et al., 2026). For each target image I_{o}, GPT-5.5 selects a rule whose applicability conditions are met and instantiates it as an image-specific degradation instruction. A proprietary image editing model then generates a candidate degraded image I_{d}. These candidates are subsequently screened through an independent verification process.

Verification, splits, and annotation. Qwen3.5-27B (Qwen, 2026a), InternVL3.5-30B-A3B (Wang et al., 2025), and GLM-4.5V (GLM-V Team et al., 2025) independently verify the generated pairs. Separate two-of-three majority votes determine whether the degradation is valid and whether the pair is suitable for automatic Face-ID comparison. Invalid degradations are discarded. Pairs suitable for automatic comparison are retained only if their FaceNet (Schroff et al., 2015) cosine similarity is at least 0.75; pairs unsuitable for automatic comparison undergo human review of identity and editing integrity. To prevent identity leakage between training and test sets, we first retrieve potential same-identity candidates using FaceNet, then form identity groups through human confirmation, ensuring that each group belongs to only one split. GPT-5.5 then annotates repair

instructions y^* for $I_{\mathrm{d}} \to I_{\mathrm{o}}$. The resulting dataset contains 12,422 pairs: 2,889 for pose, 5,397 for viewpoint, and 4,136 for framing. We reserve 200 pairs per task for evaluation and use the remaining 11,822 pairs for training.

4 METHOD

Given a portrait I_{d} and a coarse request q, PASEdit's planner generates a concrete structural editing instruction y. The executor then edits the portrait conditioned on (I_{d}, y). During training, we randomly mix factor-specified and open-ended requests at 4:1. We sample factor-specified requests from ten natural-language templates per factor (pose, viewpoint, or framing), and open-ended requests from ten templates. Both types require single-factor edits; open-ended requests also require the planner to choose the factor. Figure 3 shows the training and inference pipeline.

4.1 **AESPT**: TASK-SPECIALIZED STRUCTURAL PLANNING

Autoregressive supervised training on (I_{d}, q, y^*) yields a shared planner π_{SFT}, the common initialization and frozen reference policy for subsequent post-training. **AesPT** then proceeds in two stages: specialization and consolidation. A separate expert policy is first optimized for each structural factor using its corresponding structured feedback; the resulting capabilities are then consolidated into one planner.

Structured planning rewards. Let P, V, and F denote pose, viewpoint, and framing. For factor $t \in \{\mathrm{P}, \mathrm{V}, \mathrm{F}\}$, a frozen multimodal judge assigns each candidate plan y the reward

$$R_t(y) = \lambda_Q Q_t(I_{\mathrm{d}}, y) + \lambda_G G_t(I_{\mathrm{d}}, I_{\mathrm{o}}, y) + \lambda_A A_t(I_{\mathrm{d}}, y, y^*).$$

These terms assess planning quality, reference guidance, and image specificity. Q_t scores whether the structural edit is clear, reasonable, and executable. A_t uses the reference instruction y^* for weak directional guidance. G_t compares the candidate's improvement value on I_{d} and the target image I_{o} to distinguish image-specific plans from reusable generic advice.

Probability-weighted scoring. Following Edit-R1 (UniWorld-V2) (Li et al., 2025), we convert discrete MLLM ratings into continuous rewards. Using the full confidence distribution over ratings provides finer-grained feedback. The judge outputs logits ℓ_k for the six score tokens 0–5. We renormalize them as

$$p_k = \frac{\exp(\ell_k)}{\sum_{a=0}^{5} \exp(\ell_a)}, \quad k \in \{0, \dots, 5\}, \qquad S(p) = \frac{1}{5} \sum_{k=0}^{5} k\, p_k.$$

We obtain Q_t, A_t, and the base scores for each G_t by applying the normalized expectation $S(p)$ to their respective score-token distributions.

Image selectivity. For each candidate y, the judge separately assesses its potential to improve the degraded image I_{d} and target image I_{o} using factor-specific visual criteria. Pose concerns the subject's bodily expression. Viewpoint concerns how the camera views and presents the subject. Framing concerns the subject's arrangement within the image. These assessments yield

$$S_{\mathrm{d}}^t = S(p_{\mathrm{d}}^t), \quad S_{\mathrm{o}}^t = S(p_{\mathrm{o}}^t), \quad t \in \{\mathrm{P}, \mathrm{V}, \mathrm{F}\}.$$

If the same advice offers similar value on the already improved I_{o} and on I_{d}, it is more likely generic than specific to the current image. We thus define image selectivity as $G_t^{\mathrm{sel}} = \mathrm{clip}(S_{\mathrm{d}}^t - S_{\mathrm{o}}^t, 0, 1)$. For pose, selectivity alone cannot ensure that a plan resolves the main problem. A local hand adjustment, for example, may apply only to I_{d} without improving the overall pose. We therefore add a sufficiency score to assess whether the plan adequately resolves the main pose problem. Given (I_{d}, y), the judge uses the same probability-weighted scoring to obtain $B_{\mathrm{P}} = S(p_{\mathrm{P}}^B)$. We set $G_{\mathrm{P}} = \min\{G_{\mathrm{P}}^{\mathrm{sel}}, B_{\mathrm{P}}\}$. Viewpoint and framing directly use their selectivity scores: $G_{\mathrm{V}} = G_{\mathrm{V}}^{\mathrm{sel}}$ and $G_{\mathrm{F}} = G_{\mathrm{F}}^{\mathrm{sel}}$. We use $(\lambda_Q, \lambda_G, \lambda_A) = (0.4, 0.4, 0.2)$. The smaller λ_A keeps y^* as a soft semantic anchor without restricting the open planning space to reproducing a single reference.

Specialization and consolidation. Under these structured rewards, we optimize a separate expert policy for each factor, initialized from π_{SFT}. For rollout rewards r_i under the same input, the group mean $\bar{r}$ and standard deviation s_r give $\widehat{A}_i = (r_i - \bar{r})/s_r$ when $s_r > 0$, and zero otherwise. We

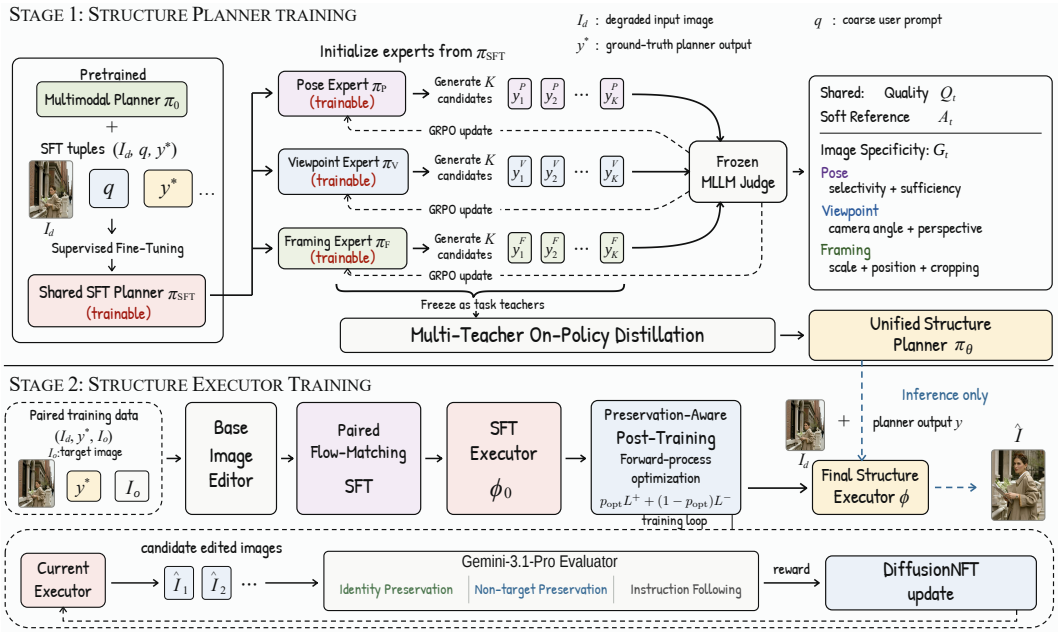


Figure 3: **Overview of PASEdit.** AesPT develops factor-specific planning capabilities and consolidates them into a unified planner, while the executor learns structural transformations and is further optimized through preservation-aware post-training via DiffusionNFT.

optimize the GRPO objective (Shao et al., 2024):

$$\mathcal{J}_{\text{GRPO}} = \mathbb{E}_{i,j}\left[\min\{\rho_{i,j}\widehat{A}_i, \text{clip}(\rho_{i,j}, 1-\epsilon, 1+\epsilon)\widehat{A}_i\} - \beta_{\text{KL}} D_{\text{KL}}(\pi_\theta \,\|\, \pi_{\text{SFT}})\right]. \tag{1}$$

Here $\rho_{i,j}$ is the current-to-old policy probability ratio at token j. Candidates have equal weight, with tokens averaged within each candidate. KL uses the same input and prefix as the ratio and keeps the policy near the shared supervised initialization. This yields specialized planners π_{P}, π_{V}, and π_{F}.

These experts are used only during training. Multi-teacher on-policy distillation (Agarwal et al., 2024; Ma et al., 2026) consolidates their capabilities into a unified student initialized from π_{SFT}. The student samples $y \sim \pi_\theta$. The frozen expert for the current factor provides next-token distributions on the student's own prefixes $h_j = (I_{\text{d}}, q, y_{<j})$. We minimize

$$\mathcal{L}_{\text{OPD}} = \mathbb{E}_{(I_{\text{d}},q,t),\, y\sim\pi_\theta}\left[\frac{1}{|y|}\sum_{j=1}^{|y|} D_{\text{KL}}\big(\pi_t(\cdot \mid h_j) \,\|\, \pi_\theta(\cdot \mid h_j)\big)\right], \tag{2}$$

where $\pi_t \in \{\pi_{\text{P}}, \pi_{\text{V}}, \pi_{\text{F}}\}$ is selected by structural factor. Supervising the student's own prefixes transfers the specialized capabilities directly into one student policy. Inference retains only the unified planner, without maintaining multiple experts.

4.2 Structural Execution

We first train an image editor on $(I_{\text{d}}, y^*, I_{\text{o}})$ using conditional flow matching (Lipman et al., 2023). It learns pose, viewpoint, and framing transformations and how to execute the planner's instructions. Given target-image latent z_0 and Gaussian noise z_1, we train velocity prediction along the interpolation path with target $v = z_1 - z_0$. This produces a base executor capable of structural editing.

We then post-train the executor for preservation while maintaining instruction adherence, adopting DiffusionNFT (Zheng et al., 2026) as the optimization algorithm. Identity preservation requires fine-grained comparisons of the person across images, so we use Gemini-3.1-Pro (Google DeepMind, 2026) as the executor's reward evaluator. For each editing condition, we generate K candidates and score identity preservation, non-target content preservation, and instruction following separately. Dividing each 0–5 rating by 5 gives R_{id}, R_{pre}, and R_{if} in $[0, 1]$, combined with equal weights as $R_{\text{edit}} = R_{\text{id}} + R_{\text{pre}} + R_{\text{if}}$. The instruction-following term discourages under-editing. Following DiffusionNFT, we use the mean reward of the same candidate group, $\mu_R = \frac{1}{K}\sum_{k=1}^{K} R_{\text{edit},k}$, as the

reference for that condition. Each candidate receives weight

$$p_{\text{opt}} = \frac{1 + \text{clip}((R_{\text{edit}} - \mu_R)/Z, -1, 1)}{2},$$

where $Z > 0$ normalizes the centered reward. We apply the forward-process objective:

$$\mathcal{L}_{\text{NFT}} = \mathbb{E}\big[p_{\text{opt}} \|v_\phi^+ - v\|_2^2 + (1 - p_{\text{opt}}) \|v_\phi^- - v\|_2^2\big], \tag{3}$$

where $v_\phi^\pm = (1 \mp \beta_{\text{NFT}}) v_{\text{old}} \pm \beta_{\text{NFT}} v_\phi$. Here v_{old} and v_ϕ are the velocity predictions of the old and current executors. The target $v = z_1 - z_0$ uses the sampled candidate's latent z_0 and Gaussian noise z_1. This stage directly updates the executor using the combined reward of generated results. It further reduces identity and non-target drift on top of the learned structural editing capability.

Inference. Inference retains only the unified planner and final executor. Given the current image and user request, the planner produces a concrete single-factor editing instruction, which the executor follows.

5 Experiments

5.1 Implementation Details

We use Qwen3.5-9B (Qwen, 2026b) and Qwen-Image-Edit-2511 (Wu et al., 2025) as the planner and executor backbones, respectively. Expert reinforcement learning uses a frozen Qwen3.5-27B as the planner reward judge, while preservation-aware executor post-training uses Gemini-3.1-Pro (Google DeepMind, 2026) as the reward evaluator. Learning rates, training schedules, and other hyperparameters are provided in Appendix B. We evaluate on 600 held-out pairs from our constructed dataset, with 200 pairs per task and no identity overlap between the training and test splits. We use GPT-5.6 Sol (OpenAI, 2026b) as the final evaluator because the evaluation requires joint reasoning over the input portrait, the editing request, and the edited result. It rates the aesthetic improvement of the requested structural factor (SI) and content preservation (CP) on a 1–5 scale. CP jointly evaluates identity and non-target content preservation, while reasonable changes in background visibility or cropping induced by the requested viewpoint or framing edit are not treated as preservation failures. Valid edit rate (VER) is the percentage of inputs scoring at least 3 on both criteria. We additionally report DeQA-Score (You et al., 2025) for image quality assessment and Q-Align (Wu et al., 2024) for image aesthetic assessment of the edited outputs; higher scores indicate better quality and aesthetics, respectively. The controlled pairs support separate evaluation of pose, viewpoint, and framing; real portraits are evaluated in Table 2 and the user study.

5.2 Main Results

Comparison methods. We compare three groups of methods with different editing capabilities. For specialized image editors, including FLUX.2-Klein-Base-9B (Black Forest Labs, 2026) and Qwen-Image-Edit-2511 (Wu et al., 2025), we prepend a Qwen3.5-9B VLM (Qwen, 2026b) as an external planner to translate the coarse request into a concrete editing instruction. Unified multimodal models, including BAGEL (Deng et al., 2025) and SenseNova-U1 (Diao et al., 2026), natively generate both a concrete editing plan and the edited image through their text and image generation branches. Proprietary models, GPT Image 2 (OpenAI, 2026c) and Nano Banana Pro (Raisinghani, 2025), are evaluated through their native interfaces. All methods in Table 1 receive the same input image, factor-specified coarse request (pose, viewpoint, or framing), and task-scope description. AesFormer performs holistic multi-factor reconstruction; we report quantitative comparisons on real portraits in Table 2 and qualitative comparisons in Section 5.5. PhotoFramer (You et al., 2026) currently releases a retrained preview model rather than the original reported weights, while the corresponding training datasets are not yet publicly available; we therefore exclude it from the quantitative comparison.

Quantitative results. On our controlled portrait editing benchmark, PASEdit achieves VERs of 92%, 88%, and 98% on pose, viewpoint, and framing, exceeding the strongest evaluated baseline for each task by 28, 52, and 4 percentage points, respectively. It also achieves the highest SI across all three tasks. Some baselines obtain high CP but very low SI, suggesting more conservative editing behavior. In contrast, PASEdit maintains high CP while making clear structural improvements, leading to higher valid edit rates. Table 1 compares the complete editing systems, while Table 3 examines the additional gains from planner and executor post-training over supervised fine-tuning on the constructed dataset.

Table 1: **Single-factor portrait structural editing.** SI: aesthetic improvement of the requested structural factor; CP: content preservation (both 1–5). VER: percentage with both scores at least 3. DeQA-Score and Q-Align assess image quality and aesthetics, respectively. Higher is better; best values are bold and second-best values are underlined, based on unrounded scores.

Method	Pose			Viewpoint			Framing			Quality	Aesthetic
	SI	CP	VER	SI	CP	VER	SI	CP	VER	DeQA	Q-Align
Specialized image editors											
Qwen-Image-Edit* (Wu et al., 2025)	2.52	4.47	44	2.01	3.74	15	1.40	4.52	6	4.21	4.65
FLUX.2-Klein* (Black Forest Labs, 2026)	2.15	4.68	31	1.78	3.97	8	1.34	4.29	4	4.20	4.67
Open-source unified models											
BAGEL (Deng et al., 2025)	2.44	4.32	37	1.86	3.94	10	2.74	2.48	8	4.18	4.67
SenseNova-U1 (Diao et al., 2026)	1.79	3.96	25	2.44	3.12	26	3.02	2.90	54	4.12	4.61
Proprietary models											
GPT Image 2 (OpenAI, 2026c)	2.48	4.66	44	2.34	**4.80**	36	4.00	4.48	94	4.29	4.77
Nano Banana Pro (Raisinghani, 2025)	3.03	4.70	64	2.49	4.02	20	1.92	**4.76**	27	4.26	4.77
PASEdit	**3.88**	**4.82**	**92**	**3.91**	4.31	**88**	**4.43**	4.64	**98**	**4.33**	**4.78**

* Uses an external VLM planner.

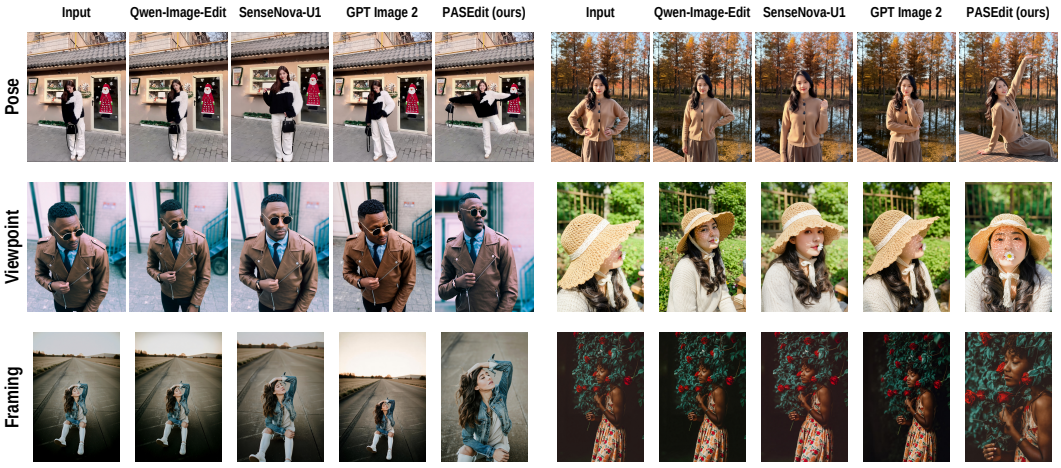


Figure 4: **Qualitative comparisons.** Two examples per task compare pose, viewpoint, and framing edits. Columns show the input, Qwen-Image-Edit, SenseNova-U1, GPT Image 2, and PASEdit.

Open-ended factor selection. We further evaluate whether the planner can identify the dominant structural factor from an open-ended request. We use the same 600 held-out test images as Table 1 and replace the factor-specified requests with randomly sampled open-ended templates. The planner achieves factor-selection accuracies of 95%, 90%, and 96% on pose, viewpoint, and framing, respectively.

As a complementary evaluation, we compare PASEdit and AesFormer on 200 real AesRecon portraits, disjoint from the 1,000 used for dataset construction. PASEdit selects one structural factor using the open-ended templates above, while AesFormer follows its original holistic reconstruction protocol. Table 2 reports GPT-5.6 Sol pairwise win rates alongside absolute DeQA-Score and Q-Align scores. AesFormer's higher Q-Align score may partly reflect tone and brightness adjustments beyond our single-factor scope.

Table 2: **Comparison with AesFormer on 200 real AesRecon portraits.** First three criteria: GPT-5.6 Sol pairwise win rates (%); gray columns: absolute DeQA-Score and Q-Align scores.

Criterion	Overall aesthetic improvement	Identity consistency	Scene content fidelity	DeQA	Q-Align
PASEdit / AesFormer	**55**/45	**73**/27	**80**/20	**4.47**/4.31	3.24/**3.51**

Qualitative results. Figure 4 shows representative comparisons across the three tasks. In these examples, PASEdit makes coordinated pose adjustments, changes the camera–subject relationship, and tightens framing while retaining recognizable clothing and scene context. Some baseline outputs make only limited structural changes or leave the targeted issue partly unresolved.

Table 3: **Training-stage ablation.** Three-task means; metrics follow Table 1. Pres. PT: preservation-aware post-training via DiffusionNFT with equally weighted rewards for identity preservation, non-target content preservation, and instruction following.

	Planner				Executor		Mean performance		
Variant	SFT	Joint RL	Expert RL	OPD	SFT	Pres. PT	SI	CP	VER (%)
(a)	—	—	—	—	—	—	1.98	4.24	22
(b)	✓	—	—	—	—	—	2.58	4.11	37
(c)	✓	—	—	—	✓	—	3.31	3.82	61
(d)	✓	✓	—	—	✓	—	3.66	3.74	71
(e)	✓	—	✓	—	✓	—	4.00	3.78	87
(f)	✓	—	✓	✓	✓	—	3.93	3.90	84
(g) Full	✓	—	✓	✓	✓	✓	**4.07**	**4.59**	**93**

5.3 Ablation Studies

Starting from a base system composed of a pretrained VLM and a pretrained editor, we progressively examine the contribution of each planner and editor training stage, reporting averages over the three tasks. For comparison, joint RL trains a single planner on mixed pose, viewpoint, and framing samples, applying the corresponding task-specific reward to each sample. Table 3 shows that planner SFT and executor SFT both substantially improve SI and VER, although CP decreases as the model makes more active structural changes. Compared with joint RL using the same task-specific rewards, separately trained factor-specific experts achieve higher SI and VER with comparable CP, supporting the benefit of task-specialized optimization. Multi-teacher OPD then consolidates the experts into a single planner, retaining most of their structural improvement capability while increasing CP from 3.78 to 3.90. Finally, preservation-aware executor post-training further raises CP to 4.59 while maintaining strong structural improvement, with the final VER reaching 93%.

5.4 User Study

To further evaluate PASEdit, we conduct a user preference study. We invite 30 participants, the vast majority of whom have AIGC-related experience, to anonymously evaluate 20 examples. These examples are disjoint from the 1,000 used for dataset construction. Each example consists of a real degraded input from AesRecon and editing results from four models under the same request. Participants select the best result along three criteria: (1) structural improvement, (2) identity consistency, and (3) non-target content preservation. Participants make a separate selection for each criterion. In total, we collect $30 \times 20 \times 3 = 1800$ votes.

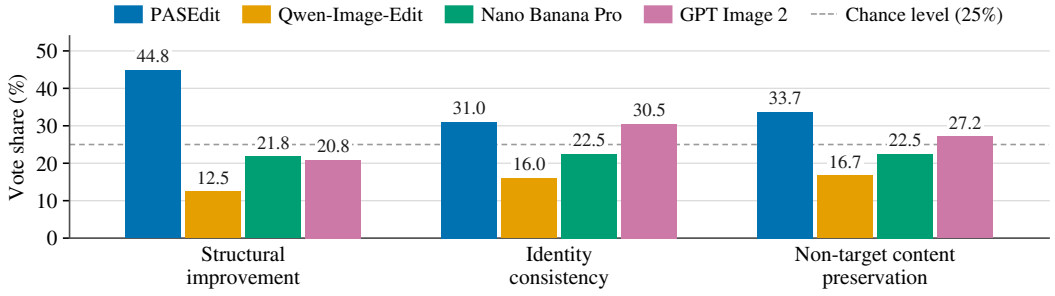


Figure 5: **User study vote shares (%).** Each criterion receives 600 votes (1800 in total). Participants choose one of four anonymized outputs per criterion.

As shown in Figure 5, PASEdit obtains the highest vote share on structural improvement and clearly outperforms the baselines. On identity consistency and non-target content preservation, PASEdit also leads or remains comparable to the strongest baseline, indicating that improving the specified structure does not come with an obvious sacrifice of identity or remaining content.

5.5 Discussion

Reward shortcuts. Without the image-specificity reward G_t, the planner tends to produce reusable editing templates that are broadly reasonable but insufficiently tailored to the current portrait. Our full structured reward introduces G_t together with planning quality and weak reference guidance to encourage image-specific planning. Figure 6 compares edits guided by plans generated

without and with G_t, showing that image-specific feedback leads to more targeted pose adjustments for the current person, clothing, and scene.



Figure 6: **Effect of the image-specificity reward G_t.** Without G_t, the planner tends to produce generic pose suggestions; adding G_t yields more targeted adjustments for the current portrait.

Structural editing and content preservation. Under the same viewpoint objective, preservation-aware executor post-training reduces identity and non-target drift while retaining the intended structural change (Figure 7). This result is consistent with Table 3, where preservation optimization substantially improves CP while maintaining strong structural improvement.



Figure 7: **Effect of preservation-aware post-training.** Both results realize the intended viewpoint change; after preservation optimization, facial appearance and background content are better preserved.

Single-factor editing versus holistic reconstruction. Complementing Table 2, Figure 8 contrasts holistic multi-factor reconstruction with targeted single-factor structural correction. AesFormer allows multiple aesthetic factors to change jointly, which in these examples is accompanied by broader changes in identity cues, scene content, depth of field, and color, while the dominant structural issue is not always addressed sufficiently or directly. In these examples, PASEdit makes more targeted corrections to the dominant structural issue and shows fewer changes to identity cues and non-target content.

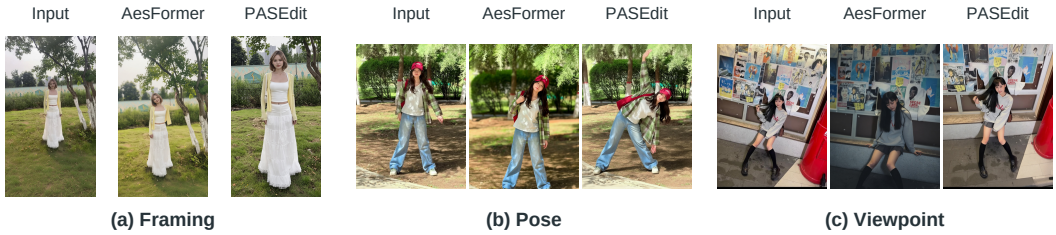


Figure 8: **Targeted single-factor correction versus holistic multi-factor reconstruction.** On real AesRecon images, each group shows the input, AesFormer, and PASEdit. PASEdit targets the dominant framing (a), pose (b), or viewpoint (c) issue.

6 Conclusion

By decoupling structural planning from image execution, PASEdit enables controllable portrait aesthetic structural editing. Combined with **AesPT** and controlled portrait editing pairs, PASEdit achieves effective single-factor improvements in pose, viewpoint, and framing while preserving subject identity and non-target content. These results show that generative portrait post-processing can extend beyond appearance adjustments to controllable structural optimization.

AI USE STATEMENT

We use generative AI for synthetic editing data construction, screening, and instruction annotation, as well as reward assessment during training and automatic evaluation, as described in Sections 3–5. AI tools also assist with the translation and language polishing of the manuscript. The authors take responsibility for the final content of this work.

ETHICS STATEMENT

The image data used in this work comprise images from AesRecon and 5,000 Unsplash portraits from PortraitCraft. These images are used for research in accordance with their applicable licenses and terms of use. The generated portraits are used solely for research and evaluation.

REPRODUCIBILITY STATEMENT

Section 3 and Appendix B describe the data construction, verification, and splitting procedures. Section 4 and Appendix A present the method and training objectives. Section 5 and Appendix B document the implementation details, training hyperparameters, and evaluation protocols.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, pp. 21246–21263, 2024.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations*, pp. 4965–4987, 2024.

Black Forest Labs. black-forest-labs/FLUX.2-klein-base-9B. Official model card, 2026. URL https://huggingface.co/black-forest-labs/FLUX.2-klein-base-9B. Accessed September 18, 2026.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, et al. Emerging properties in unified multimodal pretraining. arXiv preprint arXiv:2505.14683, 2025.

Haiwen Diao, Penghao Wu, Hanming Deng, Jiahao Wang, Shihao Bai, Silei Wu, Weichen Fan, Wenjie Ye, Wenwen Tong, Xiangyu Fan, et al. SenseNova-U1: Unifying multimodal understanding and generation with NEO-unify architecture. arXiv preprint arXiv:2605.12500, 2026.

Tianxiang Du, Hulingxiao He, and Yuxin Peng. AesFormer: Transform everyday photos into beautiful memories. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

GLM-V Team, Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, et al. GLM-4.5V and GLM-4.1V-Thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. arXiv preprint arXiv:2507.01006, 2025.

Google DeepMind. Gemini 3.1 Pro. Official model card, February 2026. URL https://deepmind.google/models/model-cards/gemini-3-1-pro/. Published February 19, 2026; accessed September 20, 2026.

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. In *International Conference on Learning Representations*, 2022.

Yipo Huang, Xiangfei Sheng, Zhichao Yang, Quan Yuan, Zhichao Duan, Pengfei Chen, Leida Li, Weisi Lin, and Guangming Shi. AesExpert: Towards multi-modality foundation model for image aesthetics perception. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 5911–5920, 2024.

Zongjian Li, Zheyuan Liu, Qihui Zhang, Bin Lin, Feize Wu, Shenghai Yuan, Zhiyuan Yan, Yang Ye, Wangbo Yu, Yuwei Niu, et al. UniWorld-V2: Reinforce image editing with diffusion negative-aware finetuning and MLLM implicit feedback. arXiv preprint arXiv:2510.16888, 2025.

Jie Liang, Hui Zeng, Miaomiao Cui, Xuansong Xie, and Lei Zhang. PPR10K: A large-scale portrait photo retouching dataset with human-region mask and group-level consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 653–661, 2021.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

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. In *Advances in Neural Information Processing Systems*, volume 38, pp. 45650–45685, 2025.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Zijie Lou, Youyun Tang, Xiaochao Qu, Haoxiang Li, Ting Liu, Luoqi Liu, Xun Zhu, Zheng Zhang, Xi Chen, Miao Li, et al. The 1st PortraitCraft challenge: A CVPR 2026 workshop competition on portrait composition understanding and generation. arXiv preprint arXiv:2606.10894, 2026.

Wenhan Ma, Jianyu Wei, Liang Zhao, Hailin Zhang, Bangjun Xiao, Lei Li, Qibin Yang, Bofei Gao, Yudong Wang, Rang Li, et al. MOPD: Multi-teacher on-policy distillation for capability integration in LLM post-training. arXiv preprint arXiv:2606.30406, 2026.

OpenAI. Introducing GPT-5.5. Official release announcement, 2026a. URL https://openai.com/index/introducing-gpt-5-5/. Accessed September 18, 2026.

OpenAI. GPT-5.6 Sol model. Official model documentation, 2026b. URL https://developers.openai.com/api/docs/models/gpt-5.6-sol. Accessed September 18, 2026.

OpenAI. GPT-Image-2 model. Official model documentation, 2026c. URL https://developers.openai.com/api/docs/models/gpt-image-2. Accessed September 18, 2026.

Qwen. Qwen/Qwen3.5-27B. Official model card, 2026a. URL https://huggingface.co/Qwen/Qwen3.5-27B. Accessed September 18, 2026.

Qwen. Qwen/Qwen3.5-9B. Official model card, 2026b. URL https://huggingface.co/Qwen/Qwen3.5-9B. Accessed September 18, 2026.

Naina Raisinghani. Introducing Nano Banana Pro. Official release announcement, 2025. URL https://blog.google/innovation-and-ai/products/nano-banana-pro/. Accessed September 18, 2026.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. FaceNet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815–823, 2015.

Yuyang Sha, Zijie Lou, Youyun Tang, Xiaochao Qu, Zheng Qu, Ben Xia, Haoxiang Li, Ting Liu, and Luoqi Liu. PortraitCraft: A benchmark for portrait composition understanding and generation. arXiv preprint arXiv:2604.03611v2, 2026.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, et al. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. arXiv preprint arXiv:2402.03300, 2024.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. arXiv preprint arXiv:2508.18265, 2025.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-Image technical report. arXiv preprint arXiv:2508.02324, 2025.

Haoning Wu, Zicheng Zhang, Weixia Zhang, Chaofeng Chen, Liang Liao, Chunyi Li, Yixuan Gao, Annan Wang, Erli Zhang, Wenxiu Sun, et al. Q-Align: Teaching LMMs for visual scoring via discrete text-defined levels. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 54015–54029, 2024.

Lianxin Xie, Wen Xue, Zhen Xu, Si Wu, Zhiwen Yu, and Hau San Wong. Blemish-aware and progressive face retouching with limited paired data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5599–5608, 2023.

Zhidan Xu, Xiaoqin Zhang, and Shijian Lu. Face retouching with diffusion data generation and spectral restorement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14722–14731, 2025.

Jiachen Yang, Xianhui Lin, Yi Dong, Zebiao Zheng, Xing Liu, Hong Gu, and Yanmei Fang. BeautyGRPO: Aesthetic alignment for face retouching via dynamic path guidance and fine-grained preference modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25110–25120, 2026.

Mingde Yao, Zhiyuan You, King-Man Tam, Menglu Wang, and Tianfan Xue. PhotoAgent: Exploratory visual aesthetic planning with large vision models. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

Zhiyuan You, Xin Cai, Jinjin Gu, Tianfan Xue, and Chao Dong. Teaching large language models to regress accurate image quality scores using score distribution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14483–14494, 2025.

Zhiyuan You, Ke Wang, He Zhang, Xin Cai, Jinjin Gu, Tianfan Xue, Chao Dong, and Zhoutong Zhang. PhotoFramer: Multi-modal image composition instruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10197–10207, 2026.

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. In *International Conference on Learning Representations*, pp. 134129–134150, 2026.

A Training Objectives

A.1 Planner SFT and GRPO

Supervised initialization minimizes the autoregressive negative log-likelihood of the reference instruction:

$$\mathcal{L}_{\mathrm{SFT}} = -\mathbb{E}_{(I_{\mathrm{d}},q,y^*)}\left[\sum_{j=1}^{|y^*|}\log \pi_\theta(y_j^* \mid I_{\mathrm{d}}, q, y_{<j}^*)\right]. \tag{4}$$

For GRPO, let $r_i = R_t(y_i)$ for N candidates sampled from the old policy under the same input $c = (I_{\mathrm{d}}, q)$. The group statistics used for the main-text advantage are

$$\bar{r} = \frac{1}{N}\sum_{k=1}^{N} r_k, \qquad s_r = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(r_k - \bar{r})^2}, \qquad \widehat{A}_i = \frac{r_i - \bar{r}}{s_r} \quad (s_r > 0). \tag{5}$$

For a group with identical rewards, the normalized advantage is assigned zero. At token j, the importance ratio is

$$\rho_{i,j}(\theta) = \frac{\pi_\theta(y_{i,j} \mid c, y_{i,<j})}{\pi_{\mathrm{old}}(y_{i,j} \mid c, y_{i,<j})}. \tag{6}$$

Writing $h_{i,j} = (c, y_{i,<j})$, Equation 1 expands to

$$\begin{aligned}\mathcal{J}_{\mathrm{GRPO}}(\theta) = \mathbb{E}\Bigg[\frac{1}{N}\sum_{i=1}^{N}\frac{1}{|y_i|}\sum_{j=1}^{|y_i|}\Big(&\min\{\rho_{i,j}\widehat{A}_i, \mathrm{clip}(\rho_{i,j}, 1-\epsilon, 1+\epsilon)\widehat{A}_i\} \\ &- \beta_{\mathrm{KL}} D_{\mathrm{KL}}\big(\pi_\theta(\cdot \mid h_{i,j}) \,\|\, \pi_{\mathrm{SFT}}(\cdot \mid h_{i,j})\big)\Big)\Bigg].\end{aligned} \tag{7}$$

The expectation covers task-specific training inputs and old-policy rollouts. Each candidate has weight $1/N$, and tokens are averaged within each candidate to account for variable-length instructions. The KL term regularizes each specialized policy toward the shared supervised initialization π_{SFT}.

A.2 Multi-Teacher On-Policy Distillation

After factor-specific GRPO, the specialized planners are frozen as teachers. The unified student is initialized from π_{SFT} and generates its own instruction $y \sim \pi_\theta(\cdot \mid I_{\mathrm{d}}, q)$. For each sample, the teacher π_t corresponding to structural factor t evaluates the student's own prefix $h_j = (I_{\mathrm{d}}, q, y_{<j})$. Let $\mathcal{V}$ denote the token vocabulary. Expanding the teacher-to-student KL in Equation 2 gives

$$\mathcal{L}_{\mathrm{OPD}} = \mathbb{E}_{(I_{\mathrm{d}},q,t),\, y\sim\pi_\theta(\cdot|I_{\mathrm{d}},q)}\left[\frac{1}{|y|}\sum_{j=1}^{|y|}\sum_{a\in\mathcal{V}}\pi_t(a \mid h_j)\log\frac{\pi_t(a \mid h_j)}{\pi_\theta(a \mid h_j)}\right]. \tag{8}$$

Here $\pi_t \in \{\pi_{\mathrm{P}}, \pi_{\mathrm{V}}, \pi_{\mathrm{F}}\}$ is selected by the structural factor. The same student parameters are updated across all factors. Both teacher and student distributions are evaluated on student-generated prefixes; sampled tokens and teacher distributions are held fixed during each update, and gradients pass only through the student's next-token probabilities.

A.3 Judge Scores and Scoring Criteria

For **AesPT**, all planner reward components use the same probability-weighted readout. Let ℓ_k denote the judge logit for score token $k \in \{0, 1, 2, 3, 4, 5\}$. We normalize only over these six tokens and compute the normalized expectation used in Section 4.1:

$$p_k = \frac{\exp(\ell_k)}{\sum_{a=0}^{5}\exp(\ell_a)}, \qquad S(p) = \frac{1}{5}\sum_{k=0}^{5} k\, p_k. \tag{9}$$

Thus $S(p) \in [0, 1]$. The same readout is used for Q_t, A_t, the base scores for factor-specific image selectivity, and pose sufficiency.

For each factor $t \in \{\mathrm{P}, \mathrm{V}, \mathrm{F}\}$, the judge separately evaluates the same candidate plan y on the degraded image I_{d} and target image I_{o}:

$$S_{\mathrm{d}}^t = S(p_{\mathrm{d}}^t), \qquad S_{\mathrm{o}}^t = S(p_{\mathrm{o}}^t), \qquad G_t^{\mathrm{sel}} = \mathrm{clip}(S_{\mathrm{d}}^t - S_{\mathrm{o}}^t, 0, 1). \tag{10}$$

If the same advice remains similarly useful on the already improved I_{o}, it is treated as less specific to the current degraded image. The factor-specific judge criteria follow the definitions in the main text: pose assesses the subject's bodily expression, viewpoint assesses how the camera views and presents the subject, and framing assesses how the subject is organized within the image.

Pose additionally uses a sufficiency score. Given (I_{d}, y), the judge evaluates whether the candidate adequately resolves the main pose problem and produces $B_{\mathrm{P}} = S(p_{\mathrm{P}}^B)$. The final image-specificity terms are

$$G_{\mathrm{P}} = \min\{G_{\mathrm{P}}^{\mathrm{sel}}, B_{\mathrm{P}}\}, \qquad G_{\mathrm{V}} = G_{\mathrm{V}}^{\mathrm{sel}}, \qquad G_{\mathrm{F}} = G_{\mathrm{F}}^{\mathrm{sel}}. \tag{11}$$

The planner reward is

$$\begin{aligned} R_t(y) &= \lambda_Q Q_t(I_{\mathrm{d}}, y) + \lambda_G G_t(I_{\mathrm{d}}, I_{\mathrm{o}}, y) + \lambda_A A_t(I_{\mathrm{d}}, y, y^*), \\ (\lambda_Q, \lambda_G, \lambda_A) &= (0.4, 0.4, 0.2). \end{aligned} \tag{12}$$

The smaller weight on A_t keeps the reference instruction as weak directional guidance rather than a unique target. Planning quality rewards clear, reasonable, and executable edits; image selectivity limits generic advice, and pose sufficiency limits local changes that leave the main pose problem unresolved. Table 4 summarizes the inputs and scoring roles.

Table 4: Judge inputs and scoring roles. Every row scores the current candidate independently.

Component	Evidence	Question addressed
Q_t	I_{d}, y	Is the proposed structural edit clear, reasonable, and executable?
A_t	I_{d}, y, y^*	Is the plan compatible with the useful direction provided by the reference instruction?
S_{d}^t	I_{d}, y	How valuable is this candidate for improving the current degraded image under factor t?
S_{o}^t	I_{o}, y	Does the same candidate remain useful on the already improved target image?
B_{P}	I_{d}, y	Does the candidate adequately resolve the main pose problem?

A.4 Paired Executor SFT

For a training triple $(I_{\mathrm{d}}, y^*, I_{\mathrm{o}})$, encode the target image as z_0, sample Gaussian noise z_1, and interpolate $z_\tau = (1 - \tau)z_0 + \tau z_1$. Conditional flow matching (Lipman et al., 2023) trains the velocity prediction toward $v = z_1 - z_0$:

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}\left[\|v_\phi(z_\tau, \tau \mid I_{\mathrm{d}}, y^*) - v\|_2^2\right]. \tag{13}$$

The expectation covers paired examples, interpolation times, and Gaussian noise.

A.5 Executor DiffusionNFT

During executor post-training, we use the ground-truth repair instruction from the dataset, setting $y = y^*$. For each editing condition $c_{\mathrm{e}} = (I_{\mathrm{d}}, y)$, the old executor generates K candidate images. Gemini-3.1-Pro independently rates identity preservation, non-target content preservation, and instruction following on a 0–5 scale. Dividing each score by 5 gives $R_{\mathrm{id},k}$, $R_{\mathrm{pre},k}$, and $R_{\mathrm{if},k}$ in $[0, 1]$, which we combine with equal weights:

$$R_{\mathrm{edit},k} = R_{\mathrm{id},k} + R_{\mathrm{pre},k} + R_{\mathrm{if},k}. \tag{14}$$

Following DiffusionNFT (Zheng et al., 2026), the reference reward for the same candidate group is

$$\mu_R = \frac{1}{K}\sum_{k=1}^{K} R_{\text{edit},k}. \tag{15}$$

Each candidate receives weight

$$p_{\text{opt},k} = \frac{1}{2} + \frac{1}{2}\,\text{clip}\left(\frac{R_{\text{edit},k} - \mu_R}{Z}, -1, 1\right). \tag{16}$$

We use $Z = 5(\sigma_R + 10^{-4})$, where σ_R is the population standard deviation of rewards within the same candidate group. Clipping maps each weight to $[0, 1]$. We omit the candidate index k for brevity. For a sampled candidate, let z_0 be its encoded latent and z_1 Gaussian noise, with

$$z_\tau = (1-\tau)z_0 + \tau z_1, \qquad v = z_1 - z_0. \tag{17}$$

The old and current executor predictions are evaluated at the same $(z_\tau, \tau \mid c_{\text{e}})$. DiffusionNFT constructs

$$\begin{aligned} v_\phi^+ &= (1-\beta_{\text{NFT}})v_{\text{old}} + \beta_{\text{NFT}} v_\phi, \\ v_\phi^- &= (1+\beta_{\text{NFT}})v_{\text{old}} - \beta_{\text{NFT}} v_\phi, \end{aligned} \tag{18}$$

and optimizes Equation 3:

$$\mathcal{L}_{\text{NFT}} = \mathbb{E}\left[p_{\text{opt}} \left\|v_\phi^+ - v\right\|_2^2 + (1 - p_{\text{opt}}) \left\|v_\phi^- - v\right\|_2^2\right]. \tag{19}$$

The expectation covers editing conditions, sampled candidates, forward-process times, and noise. The candidate weights and old-executor predictions are held fixed during each update. We set the DiffusionNFT velocity coefficient $\beta_{\text{NFT}} = 1.0$; β_{KL} denotes the separate planner KL coefficient.

B Data and Implementation Details

Data sources and verification models. The high-quality source pool combines AesRecon photographs released with AesFormer (Du et al., 2026) and 5,000 licensed Unsplash portraits (Sha et al., 2026; Lou et al., 2026). The additional Unsplash images broaden the range of subjects, scenes, and compositions. Failure-rule examples include stiff body organization, awkward limb relationships, unsuitable camera height, inappropriate subject scale, and excessive empty space. Each degradation rule is instantiated for the current portrait.

A proprietary image editing model generates candidate degraded images. Qwen3.5-27B (Qwen, 2026a), InternVL3.5-30B-A3B (Wang et al., 2025), and GLM-4.5V (GLM-V Team et al., 2025) form the verification ensemble. Each model independently assesses structural validity and whether the pair is suitable for automatic Face-ID comparison, with a separate two-of-three majority vote for each decision. Structurally invalid pairs are rejected. Among the remaining pairs, samples suitable for automatic comparison are retained only if their FaceNet (Schroff et al., 2015) cosine similarity is at least 0.75; pairs unsuitable for automatic comparison undergo human review of identity and editing integrity.

After verification, potential same-identity candidates are retrieved using FaceNet and grouped through human confirmation before splitting. All images of the same person are assigned to the same split. GPT-5.5 then annotates repair instructions y^* from the degraded input I_{d} to the original target I_{o}. Table 5 reports the resulting task counts.

Training settings. We use eight NVIDIA H20 GPUs for training. Dataset images are organized into resolution buckets around 1K, and both training and evaluation use this resolution range. For planner SFT, we fine-tune Qwen3.5-9B using AdamW (Loshchilov & Hutter, 2019) with a learning rate of 10^{-4}, an effective batch size of 32, and LoRA (Hu et al., 2022) with rank 64 and alpha 128. The SFT checkpoint at step 1,500 initializes the three factor-specific experts and serves as their shared frozen reference policy. Each expert is trained with GRPO for 1,000 steps at a learning rate of 3×10^{-6} and a batch size of 1, sampling eight candidate instructions per input and using a frozen Qwen3.5-27B as the reward judge. We set the GRPO clipping threshold to $\epsilon = 0.2$ and the KL

Table 5: Controlled-pair counts. Training counts are total minus the 200 held-out pairs in each task.

Task	Training	Test	Total
Pose	2,689	200	2,889
Viewpoint	5,197	200	5,397
Framing	3,936	200	4,136
All	11,822	600	12,422

coefficient to $\beta_{\mathrm{KL}} = 0.01$. The joint-RL baseline uses the same GRPO configuration and is trained for 3,000 steps on mixed pose, viewpoint, and framing samples, matching the total number of RL updates across the three experts. We then consolidate the experts into a unified planner through multi-teacher on-policy distillation for 600 steps at a learning rate of 10^{-5} and a batch size of 1. For executor SFT, we fine-tune Qwen-Image-Edit-2511 for 2,000 steps at a learning rate of 10^{-4}. Preservation-aware executor post-training uses DiffusionNFT for 500 steps at a learning rate of 2×10^{-5}, with eight edited images sampled per group and Gemini-3.1-Pro as the reward evaluator. Both executor SFT and RL use a batch size of 1 and LoRA with rank 32 and alpha 32.

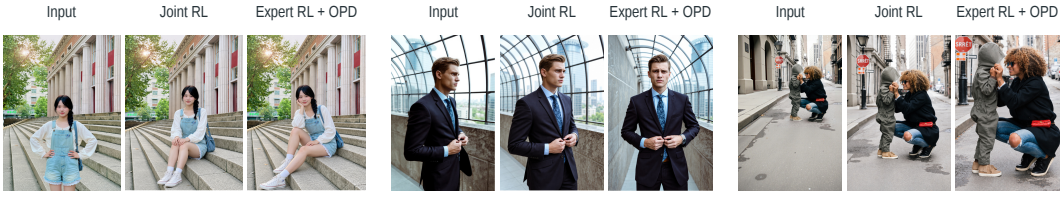
Coarse request examples. Representative factor-specified and open-ended requests include:

Pose: "Improve the person's pose."; "Make the person's pose more suitable for this portrait."
Viewpoint: "Adjust the viewpoint of this photo."; "Use a viewpoint that suits the person better."
Framing: "Improve the framing of this photo."; "Reposition the person within the frame."
Open-ended: "Improve this photo."; "Make this portrait look better."

Automatic evaluation. For the controlled benchmark in Table 1, GPT-5.6 Sol rates structural improvement (SI) and content preservation (CP) on a 1–5 scale. SI measures improvement of the requested structural factor, while CP jointly measures identity and non-target content preservation. A result contributes to the valid edit rate (VER) only when both SI and CP are at least 3. For the real-portrait comparison in Table 2, GPT-5.6 Sol instead provides pairwise preferences on overall aesthetic improvement, identity consistency, and scene content fidelity. DeQA-Score and Q-Align assess output quality and aesthetics in both tables. The final evaluator is distinct from the Gemini-3.1-Pro reward evaluator used during preservation-aware executor post-training.

C Additional Qualitative Comparison

Joint RL versus expert RL + OPD. Figure 9 compares the complete joint-RL and expert-RL-with-OPD training routes on pose, viewpoint, and framing. These examples complement the quantitative training-stage ablation in Table 3 and are not intended to isolate the effect of OPD.



(a) Pose coordination (b) Viewing direction (c) Interaction framing

Figure 9: **Joint RL versus expert RL + OPD.** Each group shows the input and the edited results produced by the two planner training routes for pose (a), viewpoint (b), and framing (c).

D VISUALIZATION RESULTS

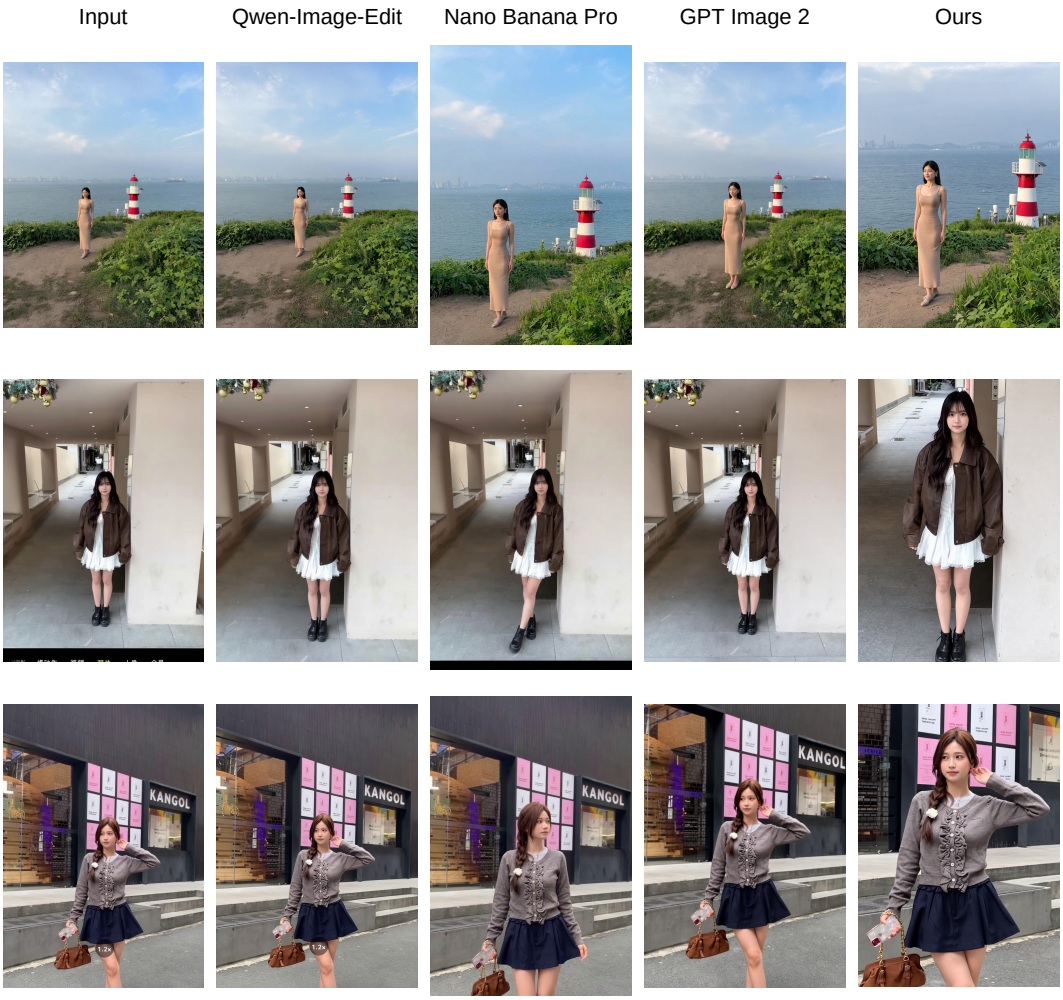


Figure 10: Framing

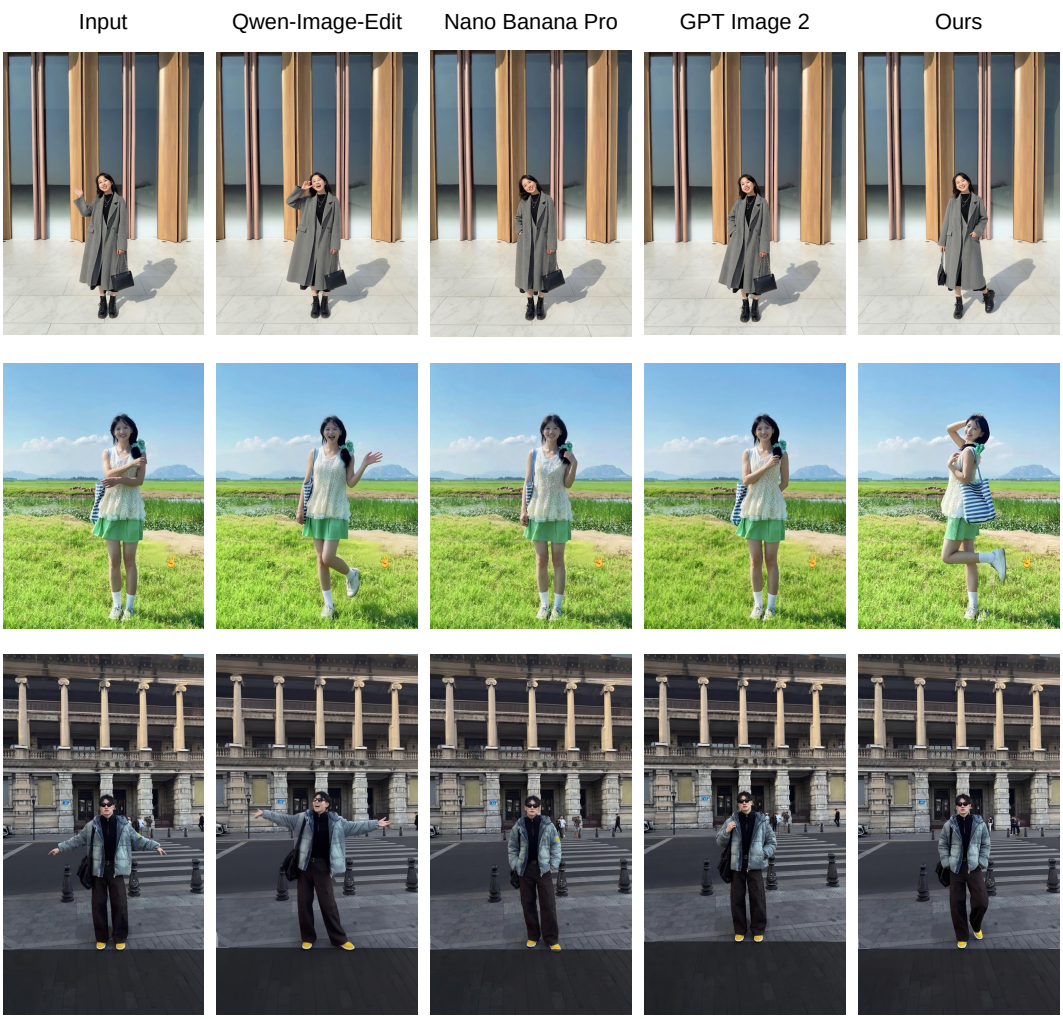


Figure 11: Pose

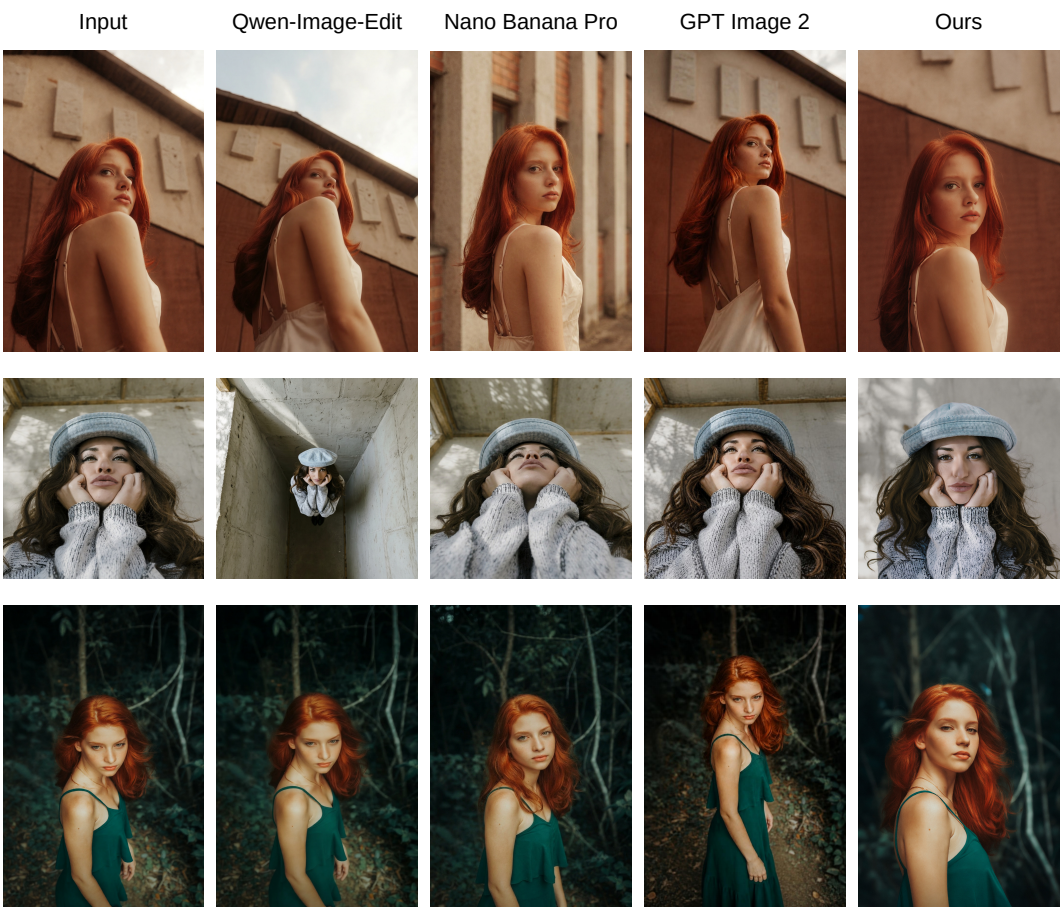


Figure 12: Viewpoint