

Edit Transfer: Learning Image Editing via Vision In-Context Relations

Lan Chen¹ Qi Mao^{1,✉} Yuchao Gu² Mike Zheng Shou²

¹MIPG, Communication University of China ²Show Lab, National University of Singapore

<https://cuc-mipg.github.io/EditTransfer.github.io>

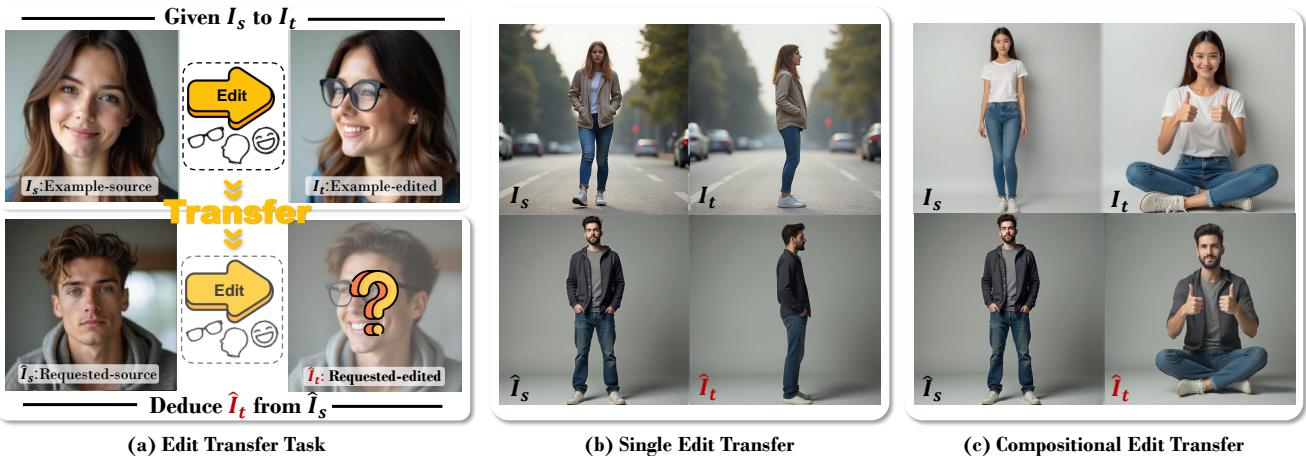


Figure 1. **Edit Transfer** aims to learn a transformation from a given source–target editing example, and apply the edit to a query image. Our framework can effectively transfer both (b) single and (c) compositional non-rigid edits via proposed visual relation in-context learning.

Abstract

We introduce a new setting, **Edit Transfer**, where a model learns a transformation from just a single source–target example and applies it to a new query image. While text-based methods excel at semantic manipulations through textual prompts, they often struggle with precise geometric details (e.g., poses and viewpoint changes). Reference-based editing, on the other hand, typically focuses on style or appearance and fails at non-rigid transformations. By explicitly learning the editing transformation from a source–target pair, **Edit Transfer** mitigates the limitations of both text-only and appearance-centric references. Drawing inspiration from in-context learning in large language models, we propose a **visual relation in-context learning** paradigm, building upon a DiT-based text-to-image model. We arrange the edited example and the query image into a unified four-panel composite, then apply lightweight LoRA fine-tuning to capture complex spatial transformations from minimal examples. Despite using only 42 training samples, **Edit Transfer** substantially outperforms state-of-the-art TIE and RIE methods on diverse non-rigid scenarios, demonstrating the effectiveness of few-shot visual relation learning.

✉ Corresponding Author

1. Introduction

Text-based image editing (TIE) has emerged as one of the most popular image generation scenarios, driven by advances in text-to-image (T2I) models [1–5]. In TIE, a user provides a source image and a textual description of the desired edit, and the model generates the edited image accordingly.

However, not all editing requirements can be easily expressed through text. For instance, as illustrated in Fig. 2(a), replicating precise arm or leg positions—such as the exact placement or jump height—can be challenging to describe purely through words. This limitation arises because while text captures abstract semantics well, it does not convey the detailed spatial information needed for non-rigid, compositional transformations.

Motivated by these limitations, we ask a simple yet intriguing question: *can we directly learn the transformation from a source image to its edited version, and then apply this transformation to a new query image?* Images naturally encode richer spatial cues than text, offering more accurate guidance for such complex manipulations.

Building on the idea of incorporating visual guidance, various reference-based image editing (RIE) methods [6–15] have been extensively explored. However, existing approaches typically focus on transferring low-level proper-

ties such as style and texture (see Fig. 2(b)), and often struggle with non-rigid spatial transformations, where more complex geometric manipulation is required.

In this paper, we propose a novel editing task, **Edit Transfer**, aiming to edit a query image by learning and applying the transformation observed in a given editing examples, as shown in Fig. 1(a). Instead of relying on large-scale training with hundreds of thousands of samples, we pose a more challenging question: *can such transformative editing be achieved using only a small set of paired images?*

Our inspiration draws from in-context learning [16] in large language models (LLMs), where a model can adapt to new tasks by observing a few example input-output pairs. By analogy, we propose a **visual relation in-context learning** for Edit Transfer: given a small number of “source image → edited image” pairs, the model is expected to transfer the demonstrated editing operation—whether it involves pose changes, facial adjustments, or other modifications—onto a new query image.

To achieve this, we build upon FLUX, a state-of-the-art text-to-image (T2I) model grounded in the DiT architecture [17], which exhibits in-context learning capabilities in image generation [18]. We adapt FLUX for visual relationship learning by embedding both the example and query images into a single four-panel composite, merging them into one token sequence. This unified representation allows the images to attend to one another through Multi-Modal Attention (MMA) [19] in the DiT blocks. Additionally, we fine-tune the Low-Rank Adaptation (LoRA) layers on a small dataset covering various editing types, strengthening the model’s capacity to capture and transfer complex visual transformations.

Remarkably, only 21 editing types—each represented by two four-panel examples—suffice to achieve effective editing transfer (Fig. 1(b)). Even more surprising, although each training sample reflects a single transformation, our model can seamlessly merge multiple editing operations present in an example pair (Fig. 1(c)), demonstrating impressive compositional abilities.

Our contributions can be summarized as follows:

- We propose a new editing task, Edit Transfer, which learns the underlying transformation from a reference example and flexibly applies it to new images, enabling edits beyond the reach of TIE and RIE methods.
- We introduce a simple yet effective visual relation in-context learning paradigm, adapting a DiT-based T2I model with minimal data (just 42 images in total), demonstrating that sophisticated editing behaviors can emerge from a handful of carefully designed examples.
- Extensive experiments validate the effectiveness of our framework, particularly in handling complex spa-

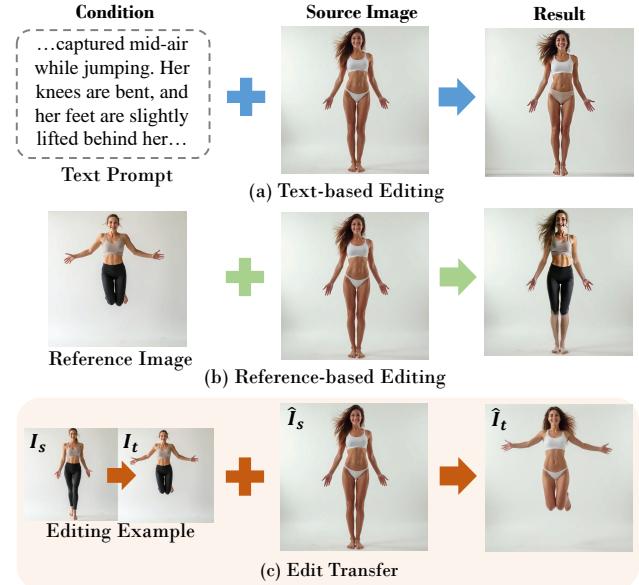


Figure 2. Comparisons with existing editing paradigms. (a) Existing TIE methods [20–25] rely solely on text prompts to edit images, making them ineffective for complex non-rigid transformations that are difficult to describe accurately. (b) Existing RIE methods [6–15] incorporate visual guidance via a reference image but primarily focus on appearance transfer, failing in non-rigid pose modifications. (c) In contrast, our proposed Edit Transfer learns and applies the transformation observed in editing examples to a query image, effectively handling intricate non-rigid edits.

tial non-rigid transformations and composite editing tasks, outperforming state-of-the-art methods in various challenging scenarios.

2. Related Work

Text-based image editing aims to modify a given image based on user-provided text that describes the desired changes. Methods such as P2P [20] and Instruct-Pix2Pix [21] successfully handle various edits, including changes in appearance, object replacement, and object addition. However, they struggle with non-rigid edits, which require large geometric variations. To address this limitation, MasaCtrl [22] provides a solution for simple non-rigid editing, such as lifting a dog’s legs to a jumping pose. Leveraging a DiT-based T2I model as backbone, recent works [23–25] achieve improved performance in handling non-rigid edits. However, for edits involving complex spatial transformations, such as human poses that engage all limbs as illustrated in Fig. 2(a), text alone often fails to provide sufficient visual guidance, leading to unsuccessful edits.

Reference-based image editing. Introducing visual guidance can compensate for the spatial details that are hard to describe with text alone, which has led to extensive research on RIE [6–15] using additional reference images.

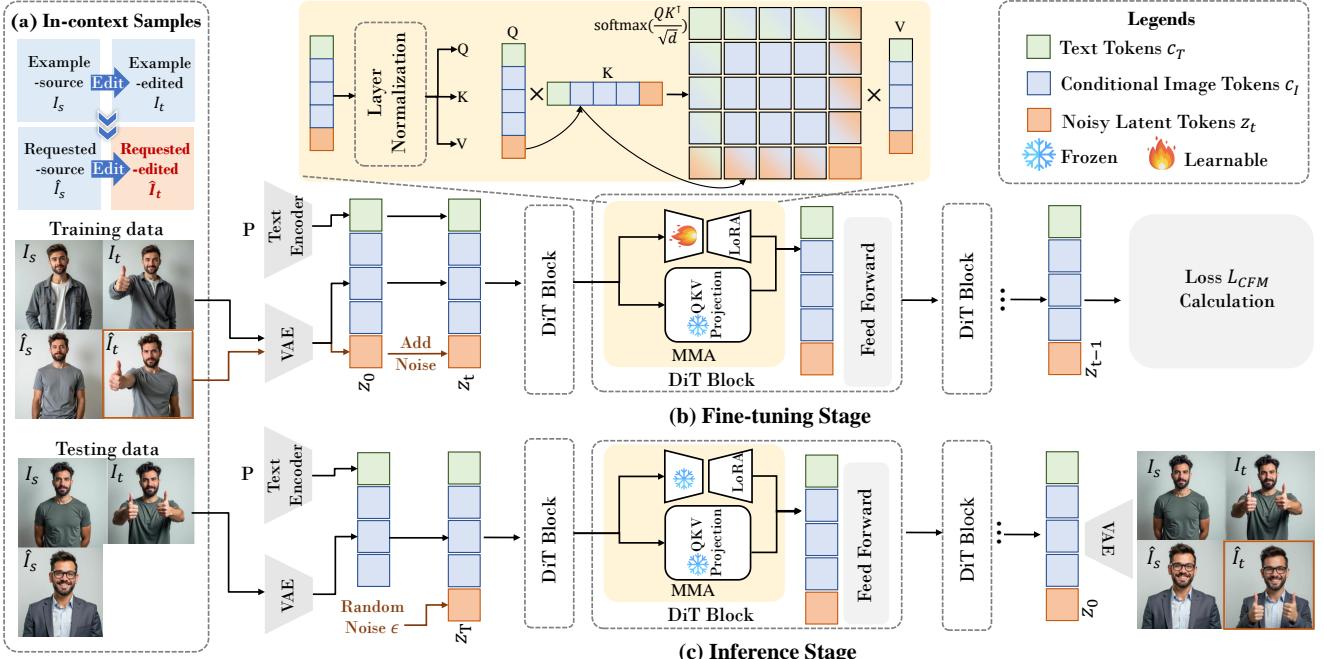


Figure 3. **Visual relation in-context learning for Edit Transfer.** (a) We arrange in-context examples in a four-panel layout: the top row (an editing pair $(\mathcal{I}_s, \mathcal{I}_t)$) and the bottom row (the query pair $(\hat{\mathcal{I}}_s, \hat{\mathcal{I}}_t)$). Our goal is to learn the transformation from $\mathcal{I}_s \rightarrow \mathcal{I}_t$, and apply it to the bottom-left image $\hat{\mathcal{I}}_s$, producing the target $\hat{\mathcal{I}}_t$, in the bottom-right. (b) We fine-tune a lightweight LoRA in the MMA to better capture visual relations. Noise addition and removal are applied only to z_t , while the conditional tokens c_T (derived from $(\mathcal{I}_s, \mathcal{I}_t, \hat{\mathcal{I}}_s)$) remain noise-free. (c) Finally, we cast Edit Transfer as an image generation task by initializing the bottom-right latent token z_T with random noise and concatenating it with the clean tokens c_I . Leveraging the enhanced in-context capability of the fine-tuned DiT blocks, the model generates \mathcal{I}_t , effectively transferring the same edits from the top row to the bottom-left image.

Early style transfer methods [6, 8] focus on transferring the global artistic style of the reference image to the target. Other approaches [7, 9] achieve appearance transfer by matching semantically aligned regions, for example, transferring textures between corresponding objects. To enable finer control, recent methods [10–15] aim to transfer the appearance of specific local regions. Despite these advances, such techniques remain confined to low-level appearance transfer and often struggle to capture intricate spatial relationships, such as pose transformations (see Fig. 2(b)). In contrast, we introduce Edit Transfer, a novel paradigm that transfers the visual relationship exemplified by an editing pair to a new query image, enabling complex spatial transformations beyond the reach of conventional RIE methods, as demonstrated in Fig. 2(c).

Vision In-context learning. In-context learning—the process of inferring patterns from provided input-output pairs—has been widely adopted in LLMs [16]. In the visual domain, in-context learning is initially explored for understanding tasks [26–29]. For instance, [29] demonstrates that in-context learning is effective for downstream representation tasks, such as segmentation. In the field of visual generation, leveraging DiT-based text-to-image (T2I) models, IC-LoRA [18] reveals that these models naturally exhibit

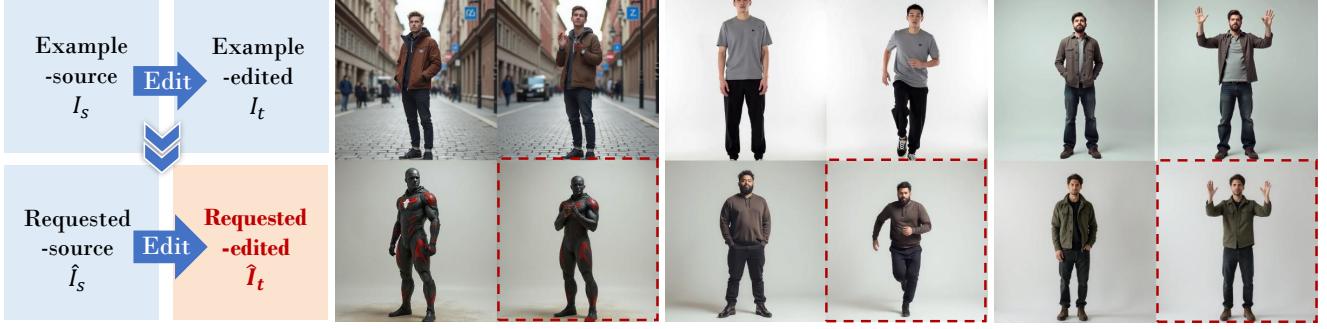
in-context generation abilities. Building on these advances, we extend the visual in-context learning paradigm to Edit Transfer in this work, thereby developing a visual relation in-context learning approach.

3. Methodology

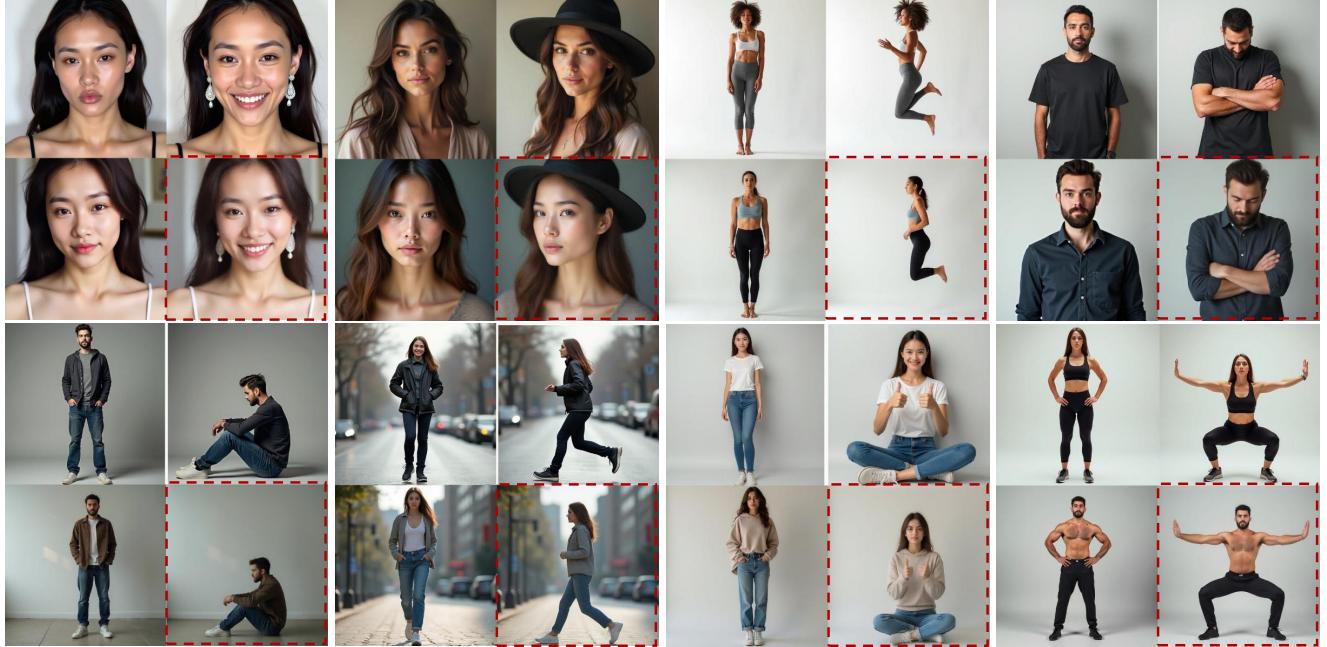
Unlike existing RIE methods that focus solely on style or appearance transfer, we introduce Edit Transfer, which learns the underlying transformation from a few-shot editing example and generalizes it to new images. As illustrated in Fig. 3(a), an editing example is defined by a pair $(\mathcal{I}_s, \mathcal{I}_t)$ —where \mathcal{I}_s is the source image and \mathcal{I}_t its edited result—and our goal is to learn the transformation from \mathcal{I}_s to \mathcal{I}_t , then apply it to a new source image $\hat{\mathcal{I}}_s$ to produce the edited image $\hat{\mathcal{I}}_t$. To this end, we first illustrate the rationale of the in-context capabilities of DiT-based T2I models in Section 3.1, and then present our visual relation in-context learning paradigm for edit transfer in Section 3.2.

3.1. Preliminary:DiT-based T2I Models

DiT-based T2I models (*e.g.*, FLUX) adopt token-based representations and transformer architectures similar to LLMs, enabling in-context generation for recent T2I systems. In these models, images are tokenized into sequences $z \in$



(a) Results of Single Edit Transfer



(b) Results of Compositional Edit Transfer

Figure 4. **Edit Transfer** exhibits impressive versatility to transfer visual exemplar pairs’ edit into the requested source image, delivering high-quality (a) single-edit transformations as well as (b) effective compositional edits that seamlessly combine multiple modifications.

$\mathbb{R}^{N \times d}$ and processed alongside text tokens $c_T \in \mathbb{R}^{M \times d}$, with their spatial and sequential structures preserved across transformer layers. In FLUX, each DiT block incorporates a MMA [19] module after layer normalization [30] to fuse image tokens z and text tokens c_T . Specifically, the concatenated sequence $[z; c_T]$ is projected into query (Q), key (K), and value (V) matrices, and MMA [19] is computed as:

$$\text{MMA}([z; c_T]) = \text{softmax} \left(\frac{QK^\top}{\sqrt{d}} \right) V. \quad (1)$$

This bidirectional attention enables image tokens to interact both among themselves and with text tokens, forming the foundation of our framework.

3.2. Visual Relation In-Context Learning

Inspired by the success of few-shot in-context learning in LLMs, we extend this approach to the visual domain by

proposing a visual relation in-context learning framework for Edit Transfer. Similar to LLMs, where carefully chosen in-context examples can greatly influence performance, we find that constructing high-quality examples is also crucial for our visual relation task. Moreover, since T2I models have not encountered this specific Edit Transfer task during pre-training, fine-tuning becomes necessary to equip them with the requisite relational understanding.

Constructing in-context samples. Unlike LLMs, which can process long token sequences, capturing relationships between separate images in FLUX requires a different approach. To encode a clear “source image \rightarrow edited image” order, we arrange each example-request pair into a 2×2 composite: the first row contains the example pair (I_s, I_t) , and the second row contains the request pair (\hat{I}_s, \hat{I}_t) . To ensure both pairs embody a consistent editing type, we synthesize multiple example pairs using the pre-trained FLUX.1-

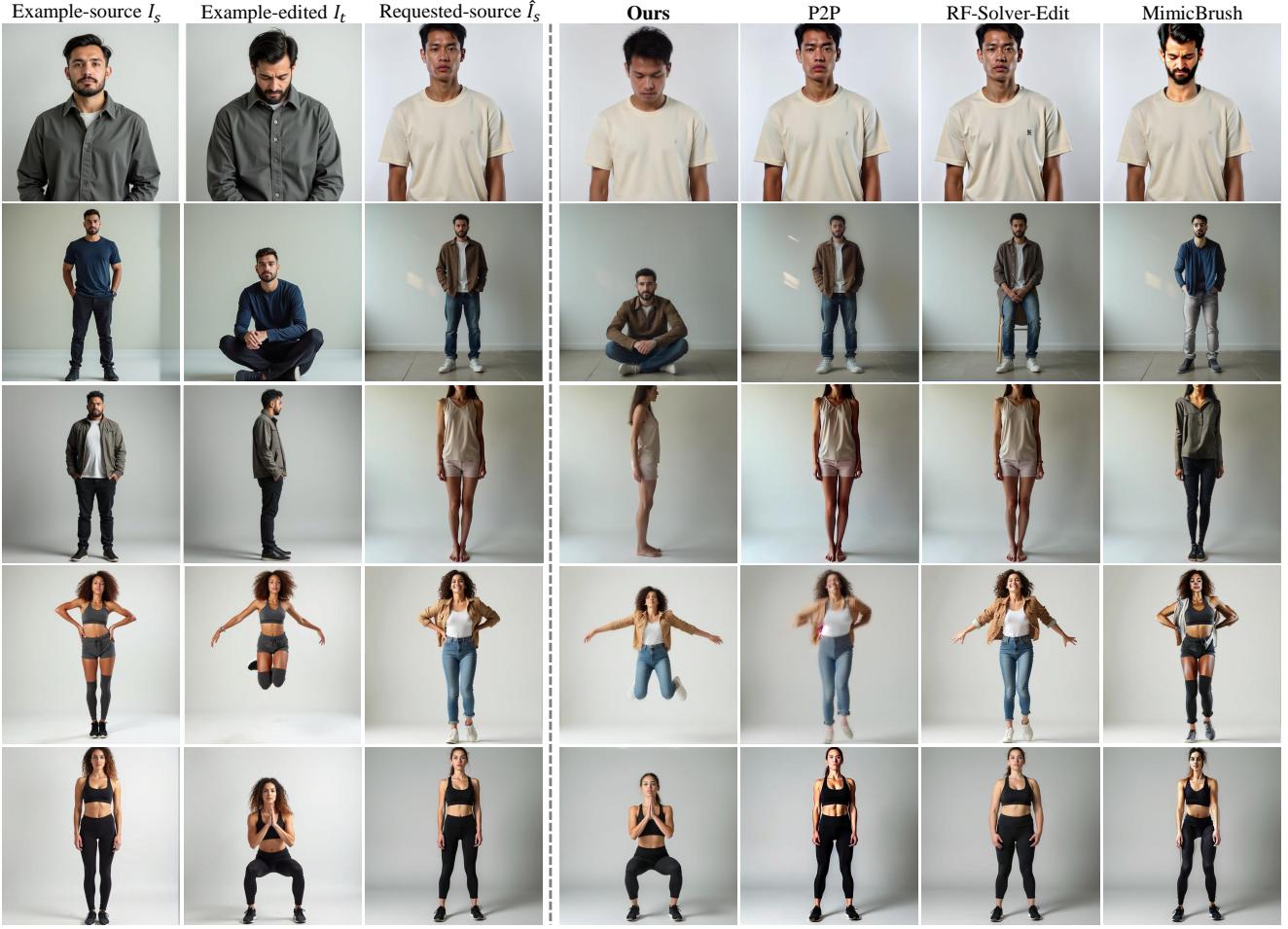


Figure 5. **Qualitative comparisons.** Compared with TIE and RIE methods, our method consistently outperforms in various non-rigid editing tasks. We provide the detailed text prompt of TIE methods in Section B.1.

dev model and manually select those with visually aligned edits. We denote the resulting four-panel composite as a training sample \mathcal{I}^* for a given editing type, as shown in Fig. 3(a). This design enables different sub-images to attend to each other via the MMA [19] module, much like how text tokens interact in LLMs. Notably, we use a simple text prompt to describe the interconnections among the four sub-images and to specify the editing type (e.g. “raising hands”), rather than providing a detailed description. We construct 21 distinct editing types for human, each represented by two diverse training examples, resulting in a dataset of 42 images. More details are provided in the Section A.1 and the effect of dataset size is discussed in Section 4.4.

Few-shot in-context fine-tuning. Building on these carefully constructed four-panel samples, our next step is to leverage them for robust in-context fine-tuning. As illustrated in Fig. 3(b), building on the four-panel training structure, our objective is to generate the bottom-right image $\hat{\mathcal{I}}_t$ conditioned on $\mathcal{I}_c = (\mathcal{I}_s, \mathcal{I}_t, \hat{\mathcal{I}}_s)$. The input consists of conditional tokens c_I (encoded from \mathcal{I}_c) and noisy tokens

z_0 (encoded from $\hat{\mathcal{I}}_t$). We add noise only to z_0 to obtain z_t , while keeping c_I tokens clean. Within each DiT block’s MMA [19] module, z_t and c_I are concatenated with c_T , and then projected into Q , K , and V . The MMA [19] module then captures the relationships among these tokens as follows:

$$\text{MMA}([z_t; c_T; c_I]) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d}}\right)V, \quad (2)$$

where $[z_t; c_T; c_I]$ denotes the concatenation of noisy latent tokens, text tokens and conditional image token. To further enhance in-context learning, we fine-tune a lightweight LoRA for the edit transfer task. The conditional flow matching loss for fine-tuning is defined as:

$$\mathcal{L}_{CFM} = E_{t, p_t(z|\epsilon), p(\epsilon)} \|v_\theta(z, t, c_T, c_I) - u_t(z|\epsilon)\|^2, \quad (3)$$

where $v_\theta(z, t, c_T, c_I)$ represents the velocity field parameterized by the model, t is the timestep, and $u_t(z|\epsilon)$ is the target vector field conditioned on noise ϵ .

Edit transfer. Empowered by the fine-tuned LoRA weights, the MMA [19] module can now transfer the relational information from a test example pair to a new source

Method	TIE		RIE	Ours
	P2P [20]	RF-Solver-Edit [23]		
CLIP-T (\uparrow)	20.22	19.83	-	22.58
CLIP-I (\uparrow)	-	-	0.804	0.810
PickScore (\uparrow)	20.60	20.96	20.34	21.50

Table 1. Quantitative comparisons of baselines and our Edit Transfer. Bold indicates the best value.

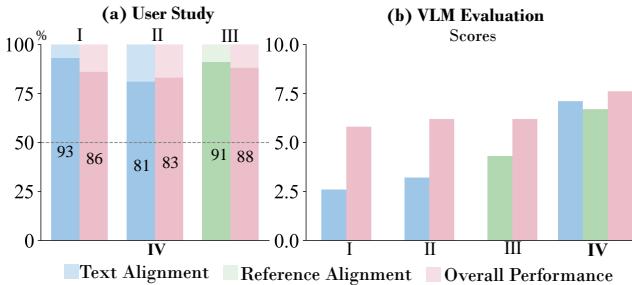


Figure 6. Results of user study and VLM evaluation. We compare our Edit Transfer (IV) with P2P [20] (I), RF-Solver-Edit [23] (II) and MimicBrush [13] (III). (a) The values show the proportion of users who prefer our method over the others. (b) The values represent the average scores given to each method by GPT-4o [31].

image \hat{I}_s . Thus, Edit Transfer is cast as an image generation task: we initialize the bottom-right image token z_T from random noise, concatenate it with c_I , and generate the output. As demonstrated in Fig. 3(c), the model successfully generates the bottom right image \hat{I}_t , effectively transferring the editing from the example pair to the new source image \hat{I}_s after denoising.

4. Experiments

4.1. Training Setup

We use FLUX.1-dev as the backbone model. During training, we set the LoRA rank to 16, the learning rate to $1e-4$, and the batch size to 4. The model is trained for 6000 iterations on a single A100 (40GB) GPU using the Adam optimizer. For more details, please refer to the Section A.2.

4.2. Experimental Results

Edit Transfer demonstrates impressive versatility by achieving both high-quality single-edit transfer and effective compositional edit transfer. As shown in Fig. 4(a), Edit Transfer consistently produces outstanding outputs for single-edit tasks: it accurately applies the transformation derived from example pairs while preserving crucial attributes, such as the identity, appearance, and background of the requested source image. Moreover, as illustrated in Fig. 4(b), Edit Transfer successfully performs compositional edit transfers in a single step, effectively combining multiple editing operations.



Figure 7. Influence of dataset scale. (a) Setting the number of training samples per editing type to $N_c = 2$ is sufficient for learning effective non-rigid edits, even when the total number of editing types $N_T = 10$. (b) Increasing $N_T = 21$ further improves the model’s ability to capture subtle local edits and enhances its generalization to cases where the editing example and query image are spatially misaligned.

4.3. Comparisons with Baselines

Baselines. We compare our method with existing representative TIE and RIE approaches.

- **TIE methods:** 1) P2P [20] is a classical text-based editing method; 2) RF-Solver-Edit [23] is a state-of-the-art editing method that incorporates MasaC-trl [22]’s principles into the DiT architecture, delivering enhanced editing capabilities.
- **RIE method:** MimicBrush [13] enables users to edit image regions by drawing on in-the-wild references without needing an exact source match.

For TIE methods, we employ GPT-4o [31] to generate detailed text prompts followed by human revision, while for RIE methods—which support only a single reference image—we use I_t as the reference.

Qualitative results. We present a visual comparison of our approach against several baselines in Fig. 5. It can be clearly observed that existing TIE methods struggle with complex spatial transformations. For instance, in the fourth row, P2P [20] fails to transfer into the expected pose. While RF-Solver-Edit [23] can perform simple non-rigid modifications, such as stretching arms, it fails to capture intricate leg movements required for a “jumping” pose. For RIE method, MimicBrush[13] focuses on transferring the texture from the reference image well, as evident in the copied clothing details in the second row of Fig. 5. However, it fails to transfer complex non-rigid transformations. In contrast, our method successfully handles complex non-rigid edits,

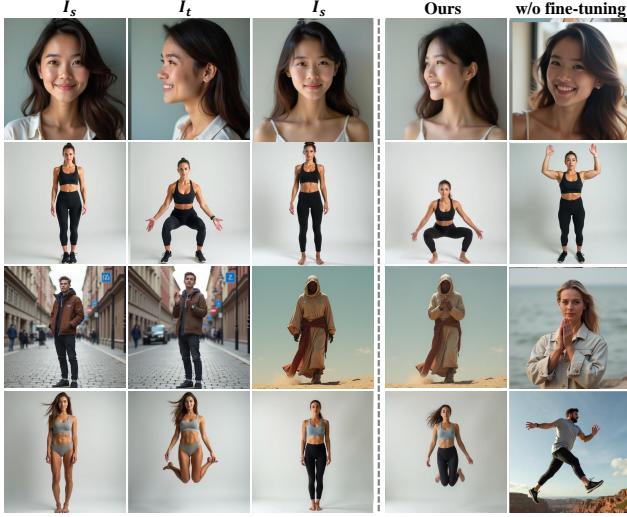


Figure 8. Ours vs. w/o fine-tuning. Without fine-tuning, Flux can only capture some of the pose information identified in I_t regardless of the relation between $I_s \rightarrow I_t$ and \hat{I}_s . In contrast, with our few-shot fine-tuning, the model effectively learns the visual relation from example pairs and applied to \hat{I}_s .

faithfully following the demonstrated examples.

Quantitative results. We quantitatively evaluate our proposed method against baseline models using automatic metrics, human evaluations, and assessments by a vision-language model (VLM). For further implementation details, please refer to the Section B.2.

Automatic Metrics. To evaluate TIE methods, we employ CLIP-T [1] to measure the alignment between text and image. For RIE method, we utilize CLIP-I [1] to quantify the similarity between the edited image \hat{I}_t and the reference image I_t . Furthermore, we incorporate PickScore [32] to assess the overall quality of the generated images. As presented in Table 1, our Edit Transfer demonstrates strong alignment with both text prompts and reference images while also achieving superior overall image quality.

User Study. We conduct a user preference study using pairwise comparisons. For text-based image editing (TIE), we evaluate text alignment to assess how well the generated images correspond to the provided descriptions. For reference-based image editing (RIE), we measure reference alignment, determining how accurately the generated images reflect the intended transformations based on the reference image. Additionally, we assess overall user preference to gauge perceived quality. As shown in Fig. 6 (a), our method achieves an outstanding preference rate exceeding 80% across all aspects, consistently outperforming the baselines in all three evaluation criteria.

VLM Evaluation. We adopt the scoring rules based on the LMM Score methodology [33] to evaluate text alignment, reference alignment, and overall performance via VLM GPT-4o [31]. Each criterion is scored on a scale from 1 to 10, with higher scores indicating better performance.

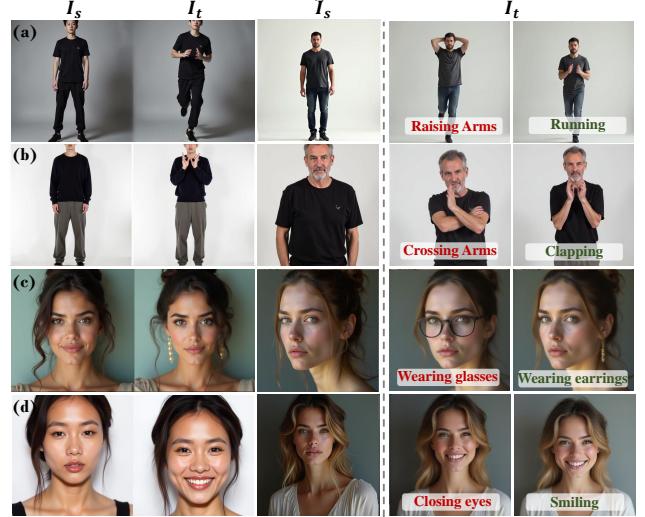


Figure 9. Investigating the alignment between text and visual example pairs. When the text prompt and visual demonstrations convey different semantics, the generated images \hat{I}_t tend to (a)(b) exhibit mixed semantics from both sources, and either (c) follow the text or (d) the visual demonstrations. Note that the red label indicates misalignment, while green label indicates alignment.

As shown in Table 1, our model achieves the highest scores against other baselines across all criterias.

4.4. Ablation Study

We perform an ablation study to investigate how in-context sample size and fine-tuning influence our visual relation in-context learning approach.

Influence of dataset scale. Effective in-context examples are crucial for Edit Transfer. To assess dataset scale, we vary the number of editing types (N_T) and examples per type (N_c). With $N_T = 10$, training with $N_c = 1$ fails to produce the desired “raising arms” transformation, whereas $N_c = 2$ succeeds (see Fig. 7(a)). Keeping $N_c = 2$ fixed, increasing N_T from 10 to 21 not only captures major non-rigid transformations but also improves fine-grained detail transfer (e.g., “wearing earrings,” as shown in Fig. 7(b)), and enhances adaptation to spatial misalignment between visual example pairs and query images. We hypothesize that, similar to the benefits of multi-task fine-tuning [34] observed in LLMs, increasing editing types diversity enhances the model’s in-context learning ability. Thus, we adopt $N_T = 21$ and $N_c = 2$ in our final configuration.

Ours vs. w/o fine-tuning. To validate the efficiency of our in-context fine-tuning strategy, we compare our model with FLUX.1-dev without fine-tuning. As shown in Fig. 8, although the model can capture some semantic aspects of the pose identified in I_t , it fails to understand the detailed relationships from $I_s \rightarrow I_t$ and the fine-grained information of I_s . For instance, while the image in the last row exhibits a “jumping” motion, the resulting posture significantly deviates from that in I_t . In contrast, our fine-tuned model



Figure 10. **Generalization performance of Edit Transfer.** Our model demonstrates remarkable generalization by: (b) Generating novel pose variations within a given editing type, even if such variations were unseen during training; (c) Flexibly combining different editing types; (d) Transferring its capabilities across other species.



Figure 11. **Limitations.** Our method struggles with low-level properties transfer e.g. color.

successfully transfers the edits demonstrated in the example pairs to the requested image \hat{I}_s , verifying that even a small set of in-context examples enables the model to learn precise visual relationships.

4.5. Discussion

Alignment between text and visual example pairs. In our framework, both text and visual example pairs guide the editing process. To evaluate their influence, we deliberately introduce a semantic mismatch between them. As shown in Fig. 9, the output may (1) blend both cues of text and visual guidance, (2) follow the text prompt, or (3) follow the visual example pairs. For instance, in Fig. 9(a), the leg pose in \hat{I}_t closely mirrors the visual demonstrations, while the arm pose follows the “raise” prompt. In Fig. 9(c) and Fig. 9(d), the output tends to follow the text prompt or visual demonstrations. These results underscore the need to align text and visual guidance for consistent editing transfer.

The generalization capability. Our model exhibits remarkable generalization. 1) It can transfer new pose styles for a training editing type, which has never been seen during training, such as different sitting styles in Fig. 10(b); 2) it can flexibly combine any poses of editing type, as illustrated in Fig. 10(c), when the source-target example pairs

include both “shouting” and “clapping” poses, the model successfully transfers both; 3) and it can even transfer edits to a different creature (e.g. panda) that is not present in the training data, as shown in Fig. 10(d).

Limitations. While our proposed framework excels in spatial and compositional editing tasks, it struggles with low-level attribute transfer. As shown in Fig. 11, the model struggles to recolor a new T-shirt based on the editing transformation demonstrated in the source–target pair.

5. Conclusions and Future Work

In this work, we introduce the novel Edit Transfer setting, where a transformation is learned directly from only a single source–target editing pair and then applied to a new image. Departing from text-centric and appearance-focused paradigms, we demonstrate that a DiT-based T2I model, equipped with visual relation in-context learning and lightweight LoRA fine-tuning, can capture complex spatial edits—even with only a handful of examples. Our results highlight the surprising potential of minimal data to yield sophisticated, non-rigid transformations once thought beyond the scope of conventional TIE and RIE approaches.

Looking ahead, Edit Transfer opens up promising directions: by gathering a small set of examples spanning various editing types, creators can easily achieve a wide range of visual effects. In future work, we aim to explore more advanced editing tasks and broaden the applicability of non-rigid editing to diverse species and scenarios.

References

- [1] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1, 7
- [2] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.
- [3] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.
- [4] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*, 2022.
- [5] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *ICML*, 2024. 1
- [6] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *CVPR*, 2016. 1, 2, 3
- [7] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 3
- [8] Yuval Alaluf, Daniel Garibi, Or Patashnik, Hadar Averbuch-Elor, and Daniel Cohen-Or. Cross-image attention for zero-shot appearance transfer. In *SIGGRAPH*, 2024. 3
- [9] Yang Zhou, Xu Gao, Zichong Chen, and Hui Huang. Attention distillation: A unified approach to visual characteristics transfer. In *CVPR*, 2025. 3
- [10] 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. In *CVPR*, 2023. 3
- [11] Songyan Chen and Jiancheng Huang. Specref: A fast training-free baseline of specific reference-condition real image editing. In *ICICML*, 2023.
- [12] 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. *arXiv preprint arXiv:2409.18071*, 2024.
- [13] Xi Chen, Yutong Feng, Mengting Chen, Yiyang Wang, Shilong Zhang, Yu Liu, Yujun Shen, and Hengshuang Zhao. Zero-shot image editing with reference imitation. In *NeurIPS*, 2025. 6, 11
- [14] Shristi Das Biswas, Matthew Shreve, Xuelu Li, Prateek Singhal, and Kaushik Roy. Pixels: Progressive image exemplar-based editing with latent surgery. In *AAAI*, 2025.
- [15] Xi Chen, Lianghua Huang, Yu Liu, Yujun Shen, Deli Zhao, and Hengshuang Zhao. Anydoor: Zero-shot object-level image customization. In *CVPR*, 2024. 1, 2, 3
- [16] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *NeurIPS*, 2020. 2, 3
- [17] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023. 2
- [18] Lianghua Huang, Wei Wang, Zhi-Fan Wu, Yupeng Shi, Huanzhang Dou, Chen Liang, Yutong Feng, Yu Liu, and Jingen Zhou. In-context lora for diffusion transformers. *arXiv preprint arXiv:2410.23775*, 2024. 2, 3
- [19] Zexu Pan, Zhaojie Luo, Jichen Yang, and Haizhou Li. Multi-modal attention for speech emotion recognition. *arXiv preprint arXiv:2009.04107*, 2020. 2, 4, 5
- [20] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. In *ICLR*, 2023. 2, 6, 11
- [21] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, 2023. 2
- [22] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yingqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *ICCV*, 2023. 2, 6
- [23] Jiangshan Wang, Junfu Pu, Zhongang Qi, Jiayi Guo, Yue Ma, Nisha Huang, Yuxin Chen, Xiu Li, and Ying Shan. Tam-ing rectified flow for inversion and editing. *arXiv preprint arXiv:2411.04746*, 2024. 2, 6, 11
- [24] Omri Avrahami, Or Patashnik, Ohad Fried, Egor Nemchinov, Kfir Aberman, Dani Lischinski, and Daniel Cohen-Or. Stable flow: Vital layers for training-free image editing. In *CVPR*, 2025.
- [25] Kunyu Feng, Yue Ma, Bingyuan Wang, Chenyang Qi, Haozhe Chen, Qifeng Chen, and Zeyu Wang. Dit4edit: Diffusion transformer for image editing. *arXiv preprint arXiv:2411.03286*, 2024. 2
- [26] Xinlong Wang, Wen Wang, Yue Cao, Chunhua Shen, and Tiejun Huang. Images speak in images: A generalist painter for in-context visual learning. In *CVPR*, 2023. 3
- [27] Amir Bar, Yossi Gandelsman, Trevor Darrell, Amir Globerson, and Alexei A. Efros. Visual prompting via image inpainting. In *NeurIPS*, 2022.
- [28] Yuanhan Zhang, Kaiyang Zhou, and Ziwei Liu. What makes good examples for visual in-context learning? In *NeurIPS*, 2023.
- [29] Yutong Bai, Xinyang Geng, Karttikeya Mangalam, Amir Bar, Alan L Yuille, Trevor Darrell, Jitendra Malik, and Alexei A Efros. Sequential modeling enables scalable learning for large vision models. In *CVPR*, 2024. 3
- [30] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 4
- [31] OpenAI. Gpt-4o system card, 2024. 6, 7, 11, 12
- [32] Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiania, Joe Penna, and Omer Levy. Pick-a-pic: an open dataset of user preferences for text-to-image generation. In *NeurIPS*, 2023. 7

- [33] Yi Huang, Jiancheng Huang, Yifan Liu, Mingfu Yan, Jiaxi Lv, Jianzhuang Liu, Wei Xiong, He Zhang, Liangliang Cao, and Shifeng Chen. Diffusion model-based image editing: A survey. *IEEE TPAMI*, 2025. [7](#), [11](#)
- [34] Bingchang Liu, Chaoyu Chen, Zi Gong, Cong Liao, Huan Wang, Zhichao Lei, Ming Liang, Dajun Chen, Min Shen, Hailian Zhou, Wei Jiang, Hang Yu, and Jianguo Li. Mft-coder: Boosting code llms with multitask fine-tuning. In *KDD*, 2024. [7](#)

Appendices

A. Implementation Details

A.1. Dataset

Leveraging its inherent in-context generation capability, we utilize FLUX.1-dev to generate diverse two-panel images, each consisting of a source image \mathcal{I}_s and a corresponding “edited” image \mathcal{I}_t . We carefully select the generated pairs to ensure subject identity consistency, that each editing pair demonstrates only a single transformation, and that the source image \mathcal{I}_s is in the front view. We manually select two pairs of the same editing type that are closely aligned in terms of viewpoint, scale, and transformation style. In total, our dataset comprises 42 images spanning 21 editing types. Fig. 12 presents one sample from each editing type in the dataset. Each four-panel image is accompanied by a text prompt that describes the relationships between the grids and roughly indicates the motion. Fig. 13 shows two prompt examples for single and compositional edit transfer.

A.2. Fine-tuning and Inference

We modify the LoRA training codes building on FLUX from the AI-Toolkit repository to align with our proposed framework. The fine-tuning process takes approximately 16 hours for 6000 iterations on a single A100 (40GB). For inference, we modify the denoising process of FLUXInpaint-Pipeline. The number of denoising steps is set to 35, and the guidance scale is set to 1.

B. Details of Comparisons with Baselines

B.1. Implementation Details of Baselines

We use the official code for P2P, RF-Solver-Edit, and MimicBrush. The masks used in MimicBrush [13] are obtained by SAM.

For P2P and RF-Solver-Edit [23], we generate prompts describing $\hat{\mathcal{I}}_s$ and $\hat{\mathcal{I}}_t$ using GPT-4o [31], with human revisions. The target prompts used by the TIE methods in Fig. 5 of the paper are presented in Fig. 14.

B.2. Evaluation Details

We utilize the CLIP-T Score, CLIP-I Score based on the CLIP ViT-L/14 model, and the Pickup Score, as imple-

```
https://github.com/black-forest-labs/flux
https://github.com/ostris/ai-toolkit
https://github.com/huggingface/diffusers/blob/
main/src/diffusers/pipelines/flux/pipeline_flux_
inpaint.py
https://github.com/google/prompt-to-prompt
https://github.com/wangjiangshan0725/RF-
Solver-Edit
https://github.com/ali-vilab/MimicBrush
https://github.com/facebookresearch/segment-
anything
```



Figure 12. **Image samples of each editing type in the dataset.**

mented in as our evaluation metrics. For VLM Score, we follow the procedure of LMM Score [33]. The evaluation process are shown in Fig. 15, t We use GPT-4o [31] to complete VLM Score.

B.3. Details of User Study

The user study comprises 198 tasks. In each task, participants are shown a source image along with two edited images: one generated by our proposed method and the other by a randomly selected baseline method. The order of the images is shuffled to ensure unbiased evaluation. A simplified text prompt is provided for comparison with P2P [20] and RF-Solver-Edit [23], while the reference image is provided for comparison with MimicBrush [13]. There are three questions for the participants to answer:

- **Text-Alignment (ours vs. text-based methods):** Which image aligns better with the [target prompt]?
- **Reference-Alignment (ours vs. reference-based methods):** Which image contains subjects that are more aligned with the [reference image] in terms of motion (or decoration)?
- **Overall Performance:** Which image do you prefer overall?

C. Additional Results

As shown in Fig. 16 and Fig. 17, our method performs well in handling complex spatial transformations. Additionally, our method generalizes effectively to compositional edit transfer, as demonstrated in Fig. 17.

```
https://github.com/showlab/loveu-tgve-2023/
blob/main/scripts/run_eval.py
```

Text Prompts

This is a four-panel image grid: [TOP-LEFT]: The original source image. [TOP-RIGHT]: An edited version of the [TOP-LEFT] image, transformed to depict a running motion. [BOTTOM-LEFT]: A second original source image. [BOTTOM-RIGHT]: An edited version of the [BOTTOM-LEFT] image, applying the same running motion transformation as used in [TOP-RIGHT].

Figure 13. Prompt example edit transfer.

Text Prompts

Row 1: a man lowers his head, facing the camera
Row 2: a sitting man with legs crossed and hands resting on his lap
Row 3: a standing woman turning right
Row 4: a jumping woman with arms extended sideways and legs bent at the knees
Row 5: a woman squats while clasping her hands together

Figure 14. Text prompts generated by GPT-4o [31] for TIE methods.

Prompt Template

Step 1: Role Play

User: I invite you to participate in an experiment on 'Evaluation of Image Editing Performance with Different Methods'. Be patient and focus. I will pay you tips according to your performance. Are you ready now?

Assistant: I'm ready and intrigued! Let's dive into the experiment. How can I assist you with the evaluation?

User: Step 1: I will provide a source image, and an editing prompt that describes the target edited image. Please first view the images and describe the source image. Then describe the editing process in simple words.

Assistant: Please go ahead and provide the source image, task indicator, and editing instruction. I'll start by describing the source image once I see it.

User: ** Source Image: **[Image]

** Prompt:** [Prompt]

Assistant: [Answer]

Step 2: Task Definition

User: Step 2: Now there are 3 edited images which are edited from the source image by 3 different methods with the same one editing prompt mentioned before. You are required to first pay enough careful examination of the 3 edited images. Then you need to score each one based on each of the four following evaluation factors separately, resulting in 2 sub-scores ranging from 1 to 10, respectively. A higher score means better.

The two factors are:

1. **Editing Accuracy**: Evaluate how closely the edited image (I2) adheres to the specified editing prompt (P), measuring the precision of the editing.

- Note that details described in P, such as motion and decorations, are important factors to consider when rating.

2. **Overall Performance**: Rate the overall quality of the edited image (I2) in terms of coherence and realism.

Before scoring, carefully specify what to be concerned in each factor in this case. This can help you score more accurately. After that, I will upload the 3 edited images one by one. When I upload one image each time, you first give a detailed analysis based on the factors.

After I upload each image, you can score them in JSON format as shown:

{ "Editing Accuracy": 2, "Overall Performance": 3 }

Assistant: [Answer]

Step 3: Scoring Process

User: [Image1]

Assistant: [Answer1]

User: [Image2]

Assistant: [Answer2]

User: [Image3]

Assistant: [Answer3]

Figure 15. Prompt template of VLM score.



Figure 16. Additional experimental results of single edit transfer.

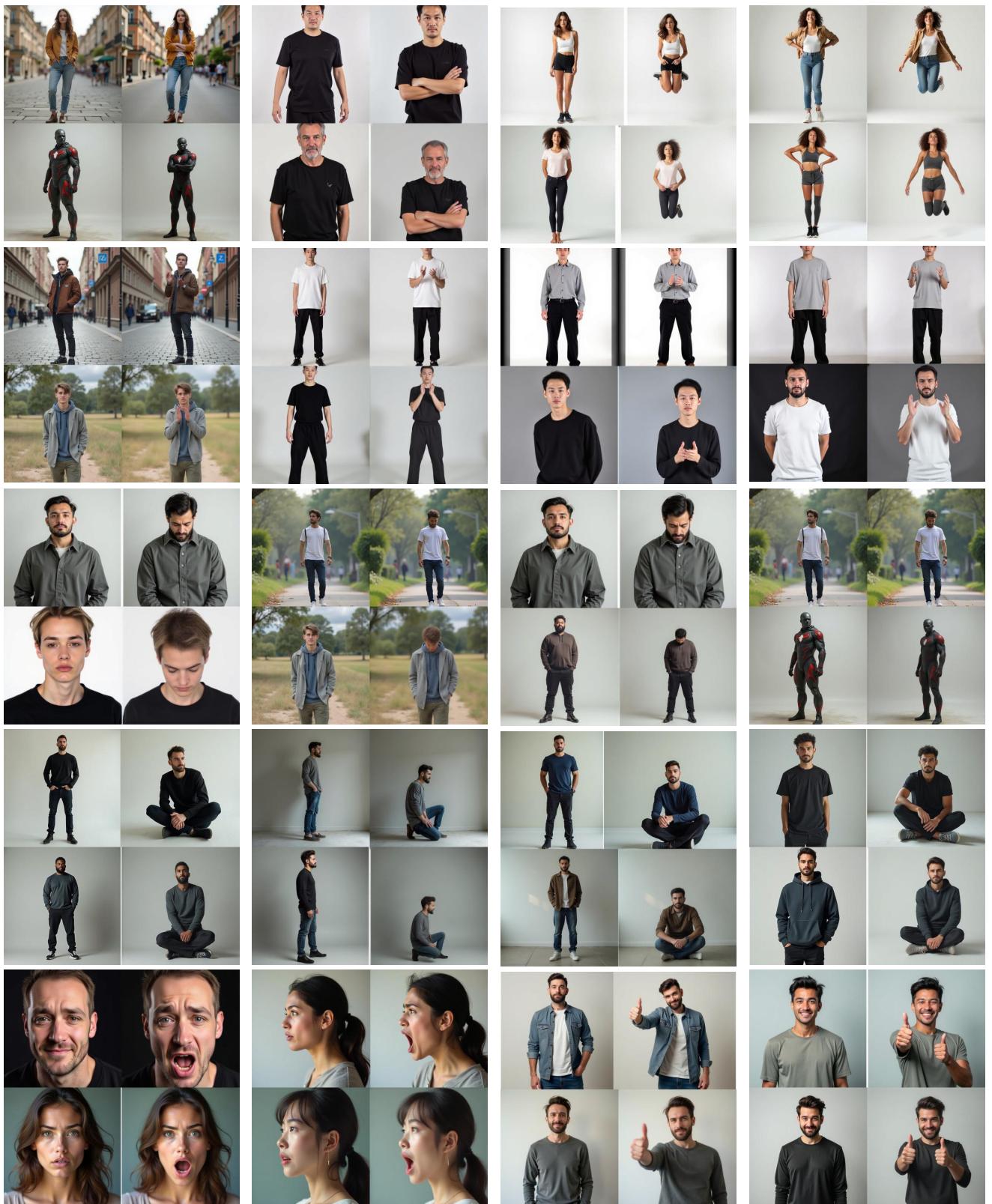


Figure 17. Additional experimental results of single edit transfer.

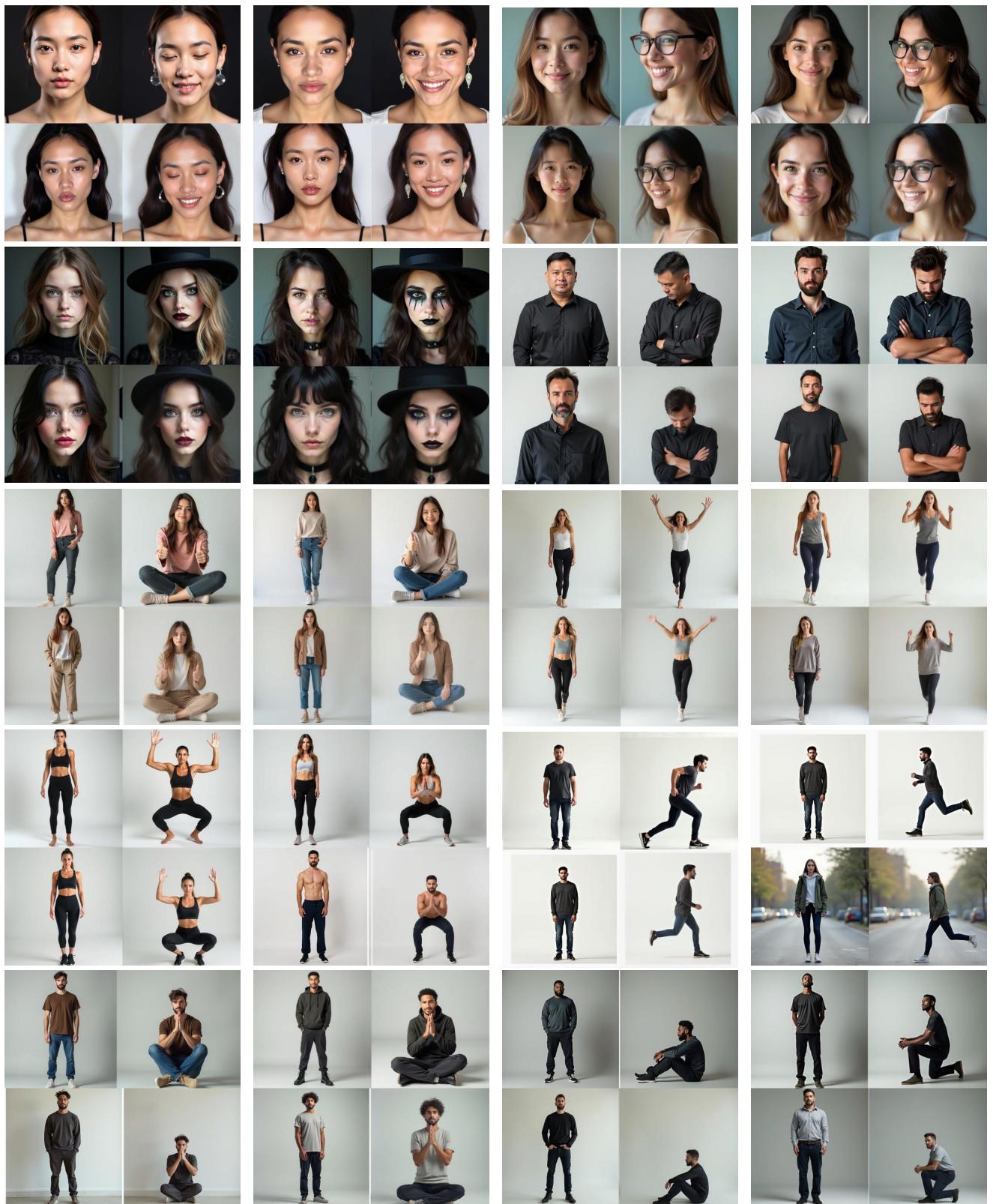


Figure 18. Additional experimental results of compositional edit transfer.