CARDBENCH-X: BENCHMARKING MEMORY-GUIDED BIDIRECTIONAL VISUAL EVOLUTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing image-generation benchmarks mainly evaluate prompt following, image quality, or the injection of new semantic knowledge. They rarely test whether an agent can separate visual knowledge that should remain invariant from knowledge that should be added or removed during an image transformation. We introduce **CardBench-X**, a benchmark for memory-guided bidirectional visual evolution. Given an input subject, a source tier, and a target tier, an agent must transform the image forward or backward across a discrete visual hierarchy while preserving identity-defining content. The benchmark evaluates target-tier realization, knowledge retention, knowledge injection, reversibility, and transfer of transformation experience across subjects and styles. We also provide **CardHarness**, a trace-driven baseline that stores successful and failed transformations, localizes over-design, under-design, identity drift, and stagnation, and retrieves validated repairs for later cases. CardBench-X offers a controlled environment for studying whether visual agents merely add more detail or actually learn reusable, direction-aware transformation knowledge.

1 INTRODUCTION

Visual improvement does not imply state control. An image editor may improve perceived quality while overshooting a requested state, moving in the wrong direction, or changing protected content. The same asset may require either a restrained or a developed presentation. We study target attainment along an ordered visual axis, including the reversal of excessive edits.

Recent experience-driven systems indicate that previous attempts can improve later generation. MemoGen stores task interpretations, references, feedback, and failure lessons, whereas GenEvolve distills contrasts between successful and unsuccessful tool trajectories into reusable visual experience Chen et al. (2026b;a). They primarily ask whether experience improves a future image. We instead define a protocol that asks whether an editing trace can transfer when presentation attributes must be added in one direction and removed or rebalanced in the other. A policy that only accumulates detail may increase a preference score while missing intentionally restrained targets; a conservative policy may preserve the input while making no meaningful progress.

Existing benchmarks emphasize prompt alignment, aesthetic preference, or instruction compliance. These criteria do not directly isolate discrete target attainment, directional asymmetry, or semantic

Table 1: Comparison with representative image-editing benchmarks. Src. denotes source-image editing, Comp. compositional or multi-turn editing, Ord. an ordinal target state, Bi-dir. explicit ascending and descending transitions, Mem. transfer from frozen training memory, and Trace trajectory-level diagnosis. A check mark indicates that the capability is part of the benchmark protocol.

Benchmark	**Editing Protocol**		**Target-State Control**			**Agent Evaluation**	
	Src.	Comp.	Ord.	Bi-dir.	Cycle	Mem.	Trace
MagicBrush Zhang et al. (2023a)	✓	✓	×	×	×	×	×
ICE-Bench Pan et al. (2025)	✓	×	×	×	×	×	×
Complex-Edit Yang et al. (2025)	✓	✓	×	×	×	×	×
CIE-Bench Bai et al. (2026)	✓	✓	×	×	×	×	×
CardBench-X (ours)	✓	✓	✓	✓	✓	✓	✓

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.

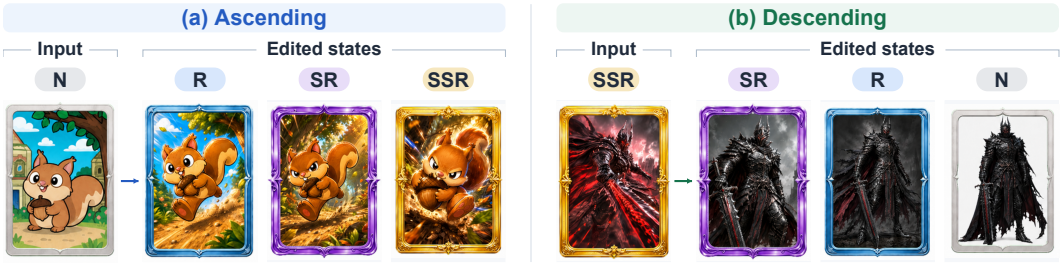

Figure 1: Bidirectional card editing with CardHarness. **Left:** the squirrel card is developed from N toward R, SR, and SSR with richer lighting, scene detail, and effects. **Right:** the armored-knight card is simplified from SSR toward SR, R, and N. Both directions aim to change the presentation state while preserving subject identity.

reversibility. We distinguish the target *visual-quality level*, defined by coordinated presentation, from *technical fidelity*: even a restrained state must reject blur, distortion, and incoherence.

Card-like illustrations provide a controlled carrier. Each subject forms a four-level ladder, N–SSR, with coordinated composition, environment, lighting, materials, effects, and detail. These are ordinal state labels, not commercial rarity predictions. Fidelity and protected-content retention remain separate validity constraints at every level.

We introduce **CardBench-X** to test this bidirectional editing contract. Its subject-specific ladders yield ascending, descending, long-gap, cycle, and held-out transfer cases. We describe a transition using protected content knowledge K_c, mutable visual-presentation knowledge K_v, and operation knowledge K_{op}. The first specifies subject attributes that should survive the edit; the second specifies the coordinated presentation attributes that define a state; the third records how those attributes should be added, removed, or rebalanced for a direction and state gap. Here, "knowledge" is an operational description of visual state and repair, not factual or IP-specific knowledge. For source and target states τ_s and τ_t, the desired operation is

$$T_{\tau_s \to \tau_t} : \left(K_c, K_v^s\right) \mapsto \left(K_c, K_v^t\right). \tag{1}$$

This formulation makes wrong-direction movement, target misses, collateral content changes, and failed inverse edits measurable without crediting generic visual enhancement as state control.

The transformation core contains 2,400 directed transitions and 2,400 cycles from 200 four-state ladders, split into 140/30/30 disjoint train/validation/test subjects. It covers both directions and gaps from one to three. Visible progress, exact landing, direction, protected-content retention, artifact safety, and semantic cycle closure are scored separately so that an editor receives no credit for merely adding ornament or leaving the source unchanged.

We also provide **CardHarness**, a transparent reference policy rather than a new generator. It diagnoses the current state, constructs a repair for the requested source–target transition, invokes a replaceable editor, validates the candidate, and records the trace. At test time, it retrieves only experience distilled from training subjects; the no-memory and shuffled-memory variants isolate the contribution of retrieval.

On the 120-case held-out evaluation, Qwen3-VL-controlled CardHarness with GPT-Image-2 reaches 77.5% valid progress, versus 59.2% for Best-of-3 under the same editor and three-call budget. This gain concerns reliable movement toward the requested state; exact landing remains substantially harder.

Table 1 compares this evaluation contract with representative image-editing benchmarks; check marks denote explicit protocol capabilities.

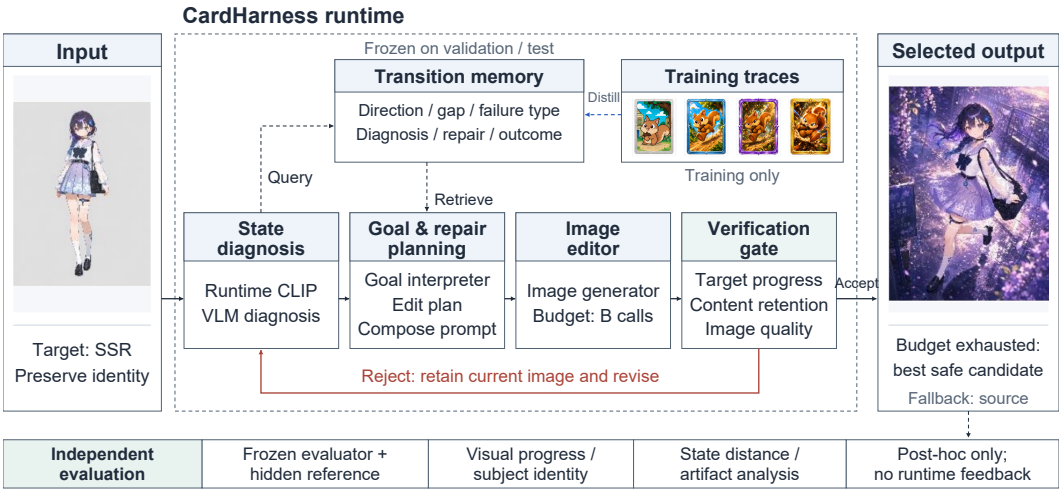

Figure 2: Overview of CardHarness. Given a source image and a target visual-quality level, the policy estimates the current state, retrieves relevant transition memory, plans an edit, and iteratively verifies the generated candidates. CardBench-X evaluates the final output independently of the generation policy.

2 Related Work

2.1 Controllable Generation and Editing

Diffusion models provide strong fidelity and prompt adherence Rombach et al. (2022); Podell et al. (2023). ControlNet and T2I-Adapter introduce spatial conditions Zhang et al. (2023b); Mou et al. (2024); DreamBooth and LoRA adapt subjects or styles Ruiz et al. (2023); Hu et al. (2022); and instruction-based methods modify content through language Brooks et al. (2023); Hertz et al. (2022). These methods specify what or where to edit, whereas CardBench-X asks how far an image should move along an ordinal visual axis and whether the same policy can move in both directions.

Editing benchmarks emphasize instruction compliance and preservation of unmodified regions. ICE-Bench spans generation and editing tasks with multi-dimensional metrics, Complex-Edit varies the number and composition of atomic instructions, and MCIE/CIE-Bench combines spatial grounding with human-filtered compliance data Pan et al. (2025); Yang et al. (2025); Bai et al. (2026). These benchmarks ask whether requested content edits are executed. CardBench-X instead specifies an ordinal end state without enumerating all edits, then tests target-state landing and semantic cycle closure. The two contracts are complementary: instruction complexity need not imply a later visual state, and a state transition may require coordinated changes that are not stated as an object-edit list.

2.2 Preference and Target-State Evaluation

CLIP similarity and preference models such as ImageReward, HPSv2, and PickScore score alignment or overall preference Radford et al. (2021); Xu et al. (2023); Wu et al. (2023b); Kirstain et al. (2023); related objectives are also used for diffusion alignment Wallace et al. (2024); Rafailov et al. (2023). These scores are monotonic, but state control is not: a restrained N output can be correct even when an aesthetic model prefers a richer image. We therefore separate target landing, direction, content retention, and output validity.

2.3 Agentic Generation and Experience Memory

Tool-oriented multimodal agents combine planning, generation, and feedback Wu et al. (2023a); Shen et al. (2023); Lian et al. (2023); Shinn et al. (2024). MemoGen stores task interpretations, references, feedback, and failure lessons; GenEvolve distills contrasts between successful and unsuccessful tool trajectories Chen et al. (2026b;a). They motivate our memory protocol but mainly evaluate future generation quality. CardBench-X instead makes the transfer of editing experience

measurable across subjects, state gaps, and reversed transition directions. CardHarness provides a reference policy for testing that transfer under a fixed generation budget.

3 CARDBENCH-X AND CARDHARNESS

CardBench-X defines the task, data, splits, and metrics. CardHarness implements one fixed-budget policy for that contract. This separation allows any image editor or agent to participate without adopting our runtime.

3.1 BIDIRECTIONAL VISUAL-EVOLUTION TASK

Let I_s be an image at source state τ_s and let $\tau_t \in \{\mathrm{N}, \mathrm{R}, \mathrm{SR}, \mathrm{SSR}\}$ be the requested target. An editing policy with optional memory $\mathcal{M}$ produces

$$\begin{aligned} \hat{I}_t &= \mathcal{G}(I_s, \tau_t; \mathcal{M}), \\ d_{state} &= |o(C_{eval}(\hat{I}_t)) - o(\tau_t)|, \end{aligned} \tag{2}$$

where C_{eval} is a frozen CLIP ordinal ranker and $o(\cdot)$ maps its output to a state index. The ranker learns only the ordering $\mathrm{N} < \mathrm{R} < \mathrm{SR} < \mathrm{SSR}$; it does not predict a commercial label or a universal aesthetic score. A valid output must also retain protected content and pass fidelity and artifact gates. Policies may query a separately trained runtime checker C_{run}, but never receive C_{eval}, its thresholds, or the target reference. This separation prevents direct reuse of the evaluation score during editing, although the two CLIP-based signals may remain correlated. N denotes a basic presentation, R a restrained development, SR a coordinated development, and SSR a showcase presentation. These names are ordinal annotations within each subject ladder, not claims about the market rarity of an existing card.

The direction is part of the task. Ascending transitions inject and coordinate the depth, lighting, material, effect, detail, and compositional cues required by the target. Descending transitions decouple, remove, or rebalance the same cues while preserving the source content and avoiding a damaged or empty image. For a semantic cycle,

$$\hat{I}_s = T_{\tau_t \to \tau_s}\big(T_{\tau_s \to \tau_t}(I_s)\big), \tag{3}$$

success requires both $C_{eval}(\hat{I}_s) = \tau_s$ and preservation of protected content. Pixel reconstruction is neither expected nor desired.

Each case contains protected content attributes A_c and mutable presentation attributes A_v. The released manifests protect face structure, hair color, skin tone, body proportion, subject category, and the broad costume silhouette. Background depth, lighting structure, material richness, effect density, detail organization, composition energy, camera, and pose remain editable. Transition knowledge K_{op} specifies how to add, remove, or rebalance A_v for a direction and gap without violating A_c. We use "knowledge" here for this operational state description, not for factual knowledge about a character or commercial card. Content retention is a benchmark constraint jointly affected by the editor and policy; it is not an identity-generation contribution of CardHarness.

3.2 DATA AND SPLITS

CardBench-X contains four variants for each of 200 prompts, with all variants of a given subject assigned to the same split. Each four-state ladder produces 12 ordered source–target pairs and 12 corresponding semantic cycles, resulting in 2,400 directed transitions and 2,400 cycles in total. For each task type, the data are divided into 1,680/360/360 cases from 140/30/30 subject-disjoint train, validation, and test groups, respectively. Across both task types, this yields 3,360/720/720 cases per split.

Although the state vocabulary is shared across subjects, target validity is defined by the requested relative transition and the corresponding subject-specific ladder. Identical state labels therefore do not imply pixel-level or stylistic equivalence across different subjects. Each ladder is constructed from a shared content specification together with four level-specific presentation profiles. The release includes the images, annotations, manifests, checksums, construction prompts, and split-generation code.

Algorithm 1 CardHarness with frozen transition memory

Require: Source image I_s, states (τ_s, τ_t), protected attributes A_c, mutable attributes A_v, memory $\mathcal{M}$, budget B
Ensure: Output I^* and execution trace $\mathcal{H}$

$I^* \leftarrow I_s$; $I^{safe} \leftarrow I_s$; $\mathcal{H} \leftarrow \emptyset$
for $b = 1$ to B **do**
  $z \leftarrow \text{Diagnose}(I^*, \tau_t, A_c, A_v)$
  $m \leftarrow \text{Retrieve}(\mathcal{M}, \tau_s, \tau_t, z)$
  $r \leftarrow \text{Repair}(z, m, A_c, A_v)$
  $\tilde{I} \leftarrow \text{Generate}(I^*, r)$
  $v \leftarrow \text{Validate}(I^*, \tilde{I}, \tau_t, A_c)$
  $\mathcal{H} \leftarrow \mathcal{H} \cup \{(z, m, r, \tilde{I}, v)\}$
  if v passes safety gates and improves final-target utility **then**
    $I^{safe} \leftarrow \tilde{I}$
  end if
  if v passes all runtime gates **then**
    $I^* \leftarrow \tilde{I}$; **break**
  else
    attach v to the next diagnosis; keep I^*
  end if
end for
if training split and the trace passes the memory-write gate **then**
  $\mathcal{M} \leftarrow \text{Distill}(\mathcal{M}, \mathcal{H})$
end if
return I^* if the target gate passed, otherwise I^{safe}; $\mathcal{H}$

Section 4 separately evaluates whether the intended ordinal structure can be recovered on held-out subjects. The construction labels are therefore treated as benchmark annotations rather than as evidence that the underlying axis is intrinsically valid.

3.3 CardHarness Policy

Each CardHarness iteration diagnoses the current state, constructs a repair for the requested source–target transition, and generates and validates a candidate. The resulting transition record can later be retrieved as memory.

Every attempt addresses the requested final state directly, and B bounds the total number of image-generation calls for the case. This direct-target policy was selected on validation data because forced adjacent traversal consumed the budget before failed edits could be repaired. The final ranker C_{eval} and the withheld target reference are absent from Algorithm 1; they are invoked only after a run terminates. On validation and test splits, $\mathcal{M}$ is read-only, so evaluated outputs cannot become later experience.

Score and diagnose. The runtime checker C_{run} estimates the ordinal state and uncertainty of the candidate, uncertainty, while the multimodal reasoner identifies which mutable attributes are inconsistent with the target. The diagnosis distinguishes a legitimate restrained presentation from a technically degraded image and records whether the previous attempt undershot, overshot, moved in the wrong direction, or damaged protected content. These signals guide the policy but do not replace the held-out evaluator used for reporting results.

Construct a repair. The reasoner combines this diagnosis with the source–target pair, protected content, target profile, and retrieved experience. It then proposes a compact repair that changes only the attributes responsible for the state mismatch. Ascending repairs add and coordinate missing presentation cues; descending repairs remove or rebalance excessive cues without treating simplification as degradation. This explicit source-to-target operation differs from unconstrained "make it better" feedback, which has no mechanism for deciding where to stop.

Generate and validate. The replaceable editor executes the repair. Validation jointly checks target movement, content retention, artifact risk, and nontrivial change. A rejected candidate is recorded but cannot become the parent of the next edit or enter memory, preventing an early failure from compounding across the trajectory. At budget exhaustion, CardHarness returns the valid candidate closest to the target, or the source if no candidate passes the safety gates. In matched comparisons, all policies share the same editor and three-call budget.

Transition memory. After a successful training case, CardHarness distills the transition type, diagnosis, repair, and validation outcome rather than storing the complete prompt. Retrieval matches the direction, state gap, and failure pattern of a new case. Memory is frozen on validation and test subjects, and retrieved experience is never recursively rewritten during evaluation. No-memory and shuffled-memory controls determine whether gains come from relevant experience rather than additional context. The resulting transition records support trajectory-level analysis of oscillation, rejected edits, and recovery after an overshoot, none of which is captured by the final score alone.

3.4 Evaluation Measures

Ordinal target distance uses the frozen CLIP ranker, which observes one image at a time and is not given the requested target or direction. A separate post-hoc evaluator R_{eval} receives the source, candidate, and a withheld same-subject target reference I_t^{ref}. It judges whether the candidate makes visible progress toward the reference state by comparing presentation attributes rather than pixels:

$$\begin{aligned} d_{state}(I,\tau) &= |o(C_{eval}(I)) - o(\tau)|, \\ P_{target}(I) &= \mathbb{1}[R_{eval}(I_s, I, I_t^{ref}) = \text{improved}], \\ A_{exact}(I) &= \mathbb{1}[C_{eval}(I) = \tau]. \end{aligned} \tag{4}$$

Progress requires visible movement in at least two target-relevant dimensions; a byte-identical no-op is deterministically rejected. *Valid progress* additionally requires content retention and raw artifact risk at most 0.45. Exact landing is a stricter secondary endpoint obtained by thresholding candidate source CLIP score to change with the requested direction and is reported separately from inferred state. Cycle success requires progress and exact state realization in the forward leg, source-state recovery in the backward leg, content retention, and artifact safety. *Strict joint success* combines valid progress with exact landing and the correct direction. Memory transfer is a paired comparison on held-out subjects against the same policy without memory and against matched shuffled retrieval. We also record image-call counts, score trajectories, and recovery behavior. Automated positive, no-op, and perturbation controls probe evaluator shortcuts and boundary sensitivity.

4 Experiments

We evaluate CardBench-X from three perspectives. We first examine whether the N–SSR annotations define a consistent ordinal axis. We then compare direct editing, iterative refinement, experience-based agents, and CardHarness under matched generation budgets. Finally, component, memory, and cycle experiments test which parts of the harness account for its behavior and where the current system still fails.

4.1 Experimental Setup

Data. CardBench-X contains 200 complete four-state ladders spanning the N, R, SR, and SSR levels. For the primary evaluation, we select 120 transitions from the subject-disjoint test split, balanced equally between ascending and descending directions. Each method receives a source image, the requested target state, and a specification of the protected content, while the target reference is reserved exclusively for evaluation. Figure 3 presents a representative benchmark ladder and illustrates the progressive visual differences across the N, R, SR, and SSR states. The recoverability of this intended ordering is evaluated quantitatively in Section 4.

Compared methods. The matched comparison fixes the GPT-Image-2 editor and a three-call budget. *GPT-Image-2(Best-of-3)* independently samples three candidates, whereas *CardHarness* uses Qwen3-VL to inspect intermediate outputs and control the same editor. GPT-5.5 is held out for final

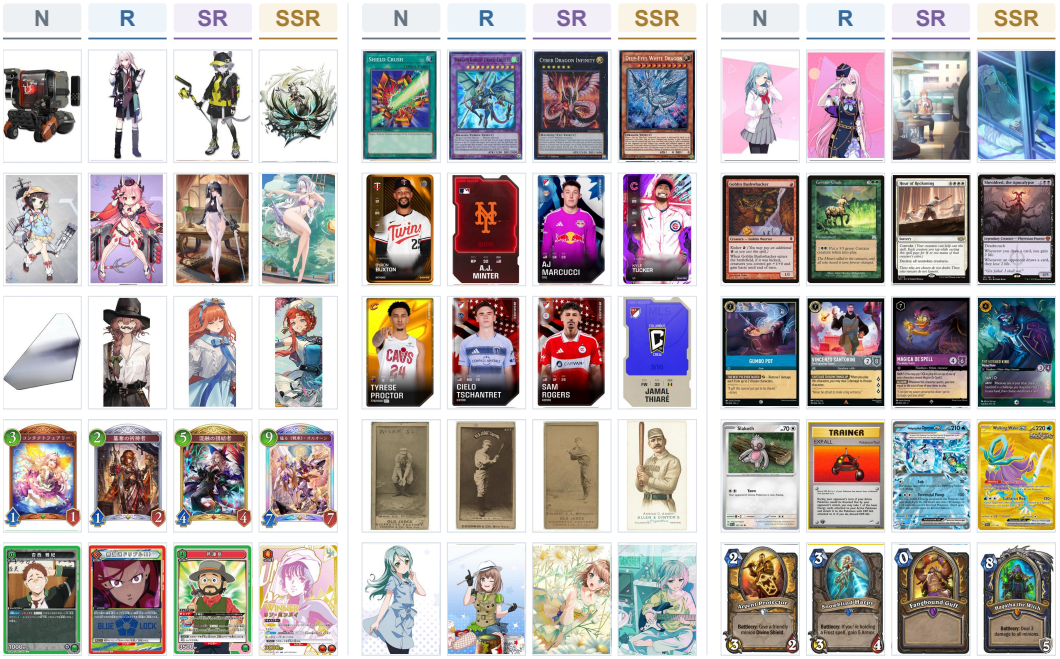

Figure 3: Representative CardBench-X benchmark ladders. Each row holds the subject identity and protected content fixed while showing the coordinated changes in composition, environment depth, lighting, material richness, and effect density from N through R and SR to SSR. The same ladders support both ascending and descending transitions; target references are hidden from methods at run time and used only for evaluation.

Table 2: Comparison of four primary systems on the 120-case test set. Calls denotes the image-generation budget, VP denotes valid progress, SJ denotes strict joint success, and Art. denotes artifact risk on a 0–100 scale.

Method	Asc. ↑	Desc. ↑	Exact ↑	Dir. ↑	VP ↑	SJ ↑	Preserve ↑	Art. ↓	Calls ↓
GPT-Image-2	51.7	46.7	**6.7**	55.0	46.7	5.0	96.7	12.6	**1**
GPT-Image-2(Best-of-3)	65.0	60.0	5.8	73.3	59.2	**5.8**	95.0	14.0	3
Mind-Brush + GPT-Image-2 He et al. (2026)	55.0	46.7	0.0	65.0	47.5	0.0	86.7	23.5	**1**
Ours	**88.3**	**70.0**	2.5	**83.3**	**77.5**	2.5	**98.3**	**12.5**	3

evaluation and is never used by either method at run time. We also report two external references that are not used for matched claims: a one-call GPT-Image-2 relay run at its native resolution and the released Mind-Brush pipeline with Qwen-Image. Additional implementation details are provided in the supplementary material.

Metrics. *Asc.* and *Desc.* measure reference-conditioned progress in the ascending and descending directions, respectively. *Exact* evaluates whether the output reaches the specified target state, while *Direction* indicates whether the ordinal score changes in the requested direction. *Preserve* measures the retention of protected content, and *Art.* reports artifact risk on a 0–100 scale. *Valid progress* (VP) requires measurable progress, preservation of protected content, and a raw artifact score below 0.45. *Strict joint success* (SJ) imposes two additional requirements: exact target-state assignment and a score change in the correct direction. For the current GPT-Image-2 comparison, an independent GPT-5.5 judge evaluates every source–output–target triplet using the same scoring protocol and criteria.

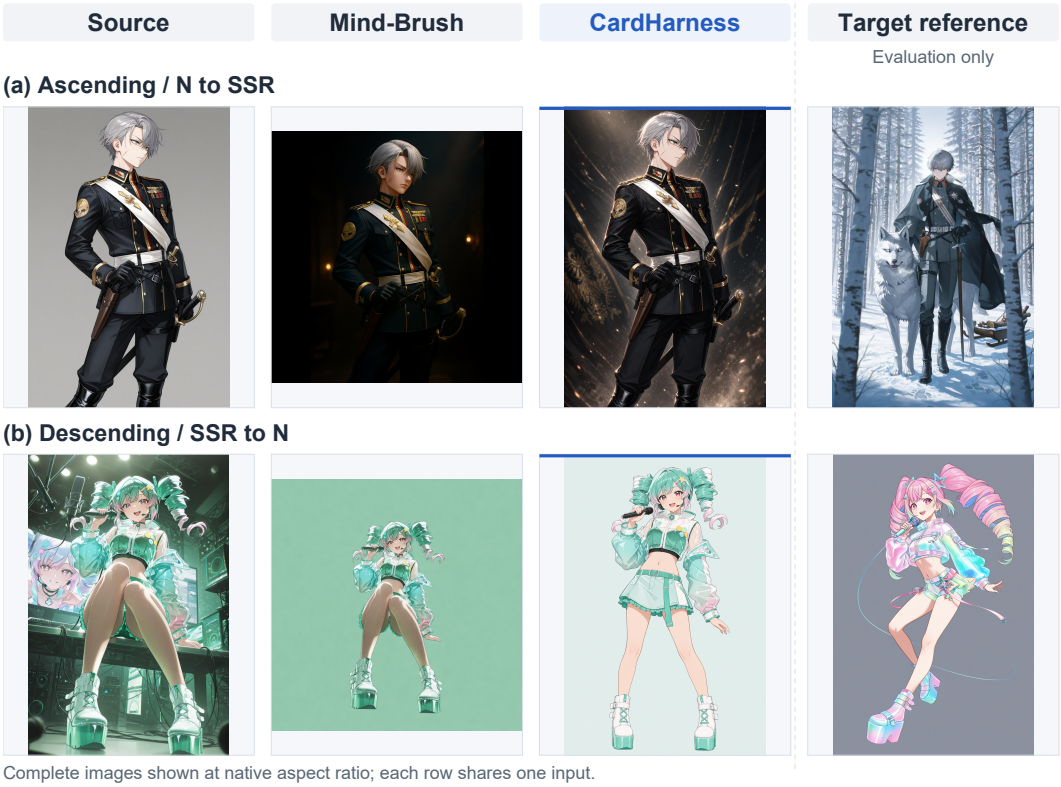

Figure 4: Bidirectional qualitative comparison between CardHarness and Mind-Brush using one input per direction. The first row shows N→SSR editing and the second shows SSR→N editing. Columns show the shared source, Mind-Brush output, CardHarness output, and hidden same-subject target reference. Complete images are fitted into equal windows without changing their aspect ratios. These selected examples illustrate editing behavior; Table 2 provides the aggregate comparison.

4.2 Benchmark Validity

The ordinal evaluator is a linear head trained on frozen CLIP ViT-L/14 image features from the training subjects. Regularization and state thresholds are selected on the validation split. The held-out results show that the benchmark captures a genuine ordering rather than only separating visually distant endpoints: accuracy is 95.56% over all ordered pairs and remains 92.22% when the comparison is restricted to adjacent states. Moreover, 76.67% of complete four-state ladders are ordered without a single inversion. Thus, the N–SSR axis is sufficiently consistent to measure whether an edit moves in the intended direction, including the difficult neighboring transitions. More information can be found in the appendix.

4.3 Main Comparison

Table 2 compares four systems on the same 120 cases. With GPT-Image-2 and three image calls fixed, CardHarness improves mean progress from 62.5% to 79.2% and valid progress from 59.2% to 77.5% over Best-of-3. This comparison measures adaptive inspection and validation under a matched image-call budget; it does not equate total inference compute. Ascending progress rises from 65.0% to 88.3% and descending progress from 60.0% to 70.0%. Direction success increases from 73.3% to 83.3%, protected-content retention from 95.0% to 98.3%, and artifact risk decreases from 14.0 to 12.5.

Exact landing remains a limitation: CardHarness attains 2.5%, versus 5.8% for Best-of-3. More reliable movement therefore does not imply more precise target attainment. The one-call relay and Mind-Brush results provide cross-setting context, not evidence for the matched comparison. Likewise, the earlier frozen block in the supplement shows stronger VLM than scalar refinement and

Table 3: Component ablations and frozen-memory controls evaluated on 60 cases per variant. VP denotes valid progress, Dir. denotes direction success, Preserve denotes protected-content retention, and Art. denotes artifact risk on a 0–100 scale. Bold indicates the best result within each group.

Variant	VP ↑	Dir. ↑	Preserve ↑	Art. ↓
CardHarness	**76.7**	65.0	96.7	11.7
w/o tier profiles	**76.7**	60.0	98.3	9.1
w/o VLM feedback	58.3	16.7	**100.0**	**7.4**
w/o validation/rejection	71.7	61.7	95.0	12.6
w/o route selection	71.7	**71.7**	91.7	14.8
No memory	**81.7**	**71.7**	95.0	13.0
Prompt-only memory	71.7	63.3	**98.3**	**11.8**
Route-only memory	78.3	68.3	95.0	13.9
Transition memory	80.0	70.0	96.7	13.3
Shuffled transition memory	76.7	68.3	96.7	13.8

high valid progress for native GPT-Image editors, but differences between blocks preclude direct attribution to the harness. Figure 4 illustrates selected bidirectional edits; these examples do not replace aggregate evaluation.

4.4 Component, Memory, and Cycle Analysis

Table 3 reports 60 cases per variant. Removing VLM feedback causes the largest drop: valid progress falls from 76.7% to 58.3% and direction success from 65.0% to 16.7%. Perfect preservation and low artifact risk in this variant reflect conservative outputs that often fail to edit, motivating joint assessment of progress and preservation.

Removing validation or route selection reduces valid progress by five points. Without routing, preservation also falls from 96.7% to 91.7% and artifact risk rises. Removing tier profiles leaves valid progress unchanged and lowers direction success by five points. These results support VLM feedback as the central component, with validation and routing as safeguards, but do not show that every module is individually necessary.

Memory provides a negative result: frozen transition memory achieves 80.0% valid progress, below the no-memory control's 81.7%; prompt-only memory reaches 71.7%. Thus, the current implementation does not demonstrate reliable cross-subject transfer. Its main gains are attributable to within-case diagnosis, validation, and repair rather than stored experience.

Only 3 of 60 forward–backward trajectories satisfy the complete cycle contract. Population-level movement in both directions therefore does not yet translate into reliable instance-level reversibility.

5 Conclusion

CardBench-X separates visual enhancement from explicit target state control. Its subject-disjoint transformation assesses whether an editor can move the same content along an ordinal presentation axis in either direction to achieve the desired state and restore the original state through reverse editing. CardHarness provides a model-independent test-time strategy for this setting, combining state diagnostics, candidate state validation, and rollback under a fixed generation budget. Results show that this structured control improves the reliability of effective editing, outperforming independent sampling, but accurately achieving the target state remains challenging. Current benchmarks are limited to a single labeled presentation axis and card-based illustrations, rather than general visual quality metrics. Therefore, their ordinal labels should be interpreted within this specific evaluation framework. Extending benchmarks to a broader visual domain and a richer state vocabulary is an important direction for future work.

AI USE STATEMENT

Generative AI was used for image generation in this work. The image-editing, runtime visual-language feedback, and automated evaluation components are described in Sections 3 and 4, with prompts and information boundaries documented in the supplementary material. These model outputs and judgments are components of the reported experimental protocol, not independent human verification. An AI assistant also helped edit manuscript wording and LaTeX formatting during submission preparation. Beyond the uses described here, no additional generative-AI assistance was used for research ideas, experimental design, method implementation, translation, or result interpretation. The authors take responsibility for the final text, claims, results, and artifacts, including content produced with generative AI.

REFERENCES

Xuehai Bai, Xiaoling Gu, Akide Liu, Hangjie Yuan, Yifan Zhang, and Jack Ma. Mcie: Multimodal llm-driven complex instruction image editing with spatial guidance. *arXiv preprint arXiv:2602.07993*, 2026.

Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18392–18402, 2023.

Sixiang Chen, Zhaohu Xing, Tian Ye, Xinyu Geng, Yunlong Lin, Jianyu Lai, Xuanhua He, Fuxiang Zhai, Jialin Gao, and Lei Zhu. Genevolve: Self-evolving image generation agents via tool-orchestrated visual experience distillation. *arXiv preprint arXiv:2605.21605*, 2026a.

Wenshuo Chen, Kuimou Yu, Bowen Tian, Jianfei Song, Shaofeng Liang, Haozhe Jia, Kan Cheng, Haosen Li, Kaishen Yuan, Lei Wang, Jiemin Wu, Songning Lai, and Yutao Yue. Memogen: Can past experience improve future text-to-image generation? *arXiv preprint arXiv:2606.03243*, 2026b.

Jun He, Junyan Ye, Zilong Huang, Dongzhi Jiang, Chenjue Zhang, Leqi Zhu, Renrui Zhang, Xiang Zhang, and Weijia Li. Mind-brush: Integrating agentic cognitive search and reasoning into image generation. *arXiv preprint arXiv:2602.01756*, 2026.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *Advances in neural information processing systems*, 36:36652–36663, 2023.

Long Lian, Boyi Li, Adam Yala, and Trevor Darrell. Llm-grounded diffusion: Enhancing prompt understanding of text-to-image diffusion models with large language models. *arXiv preprint arXiv:2305.13655*, 2023.

Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 4296–4304, 2024.

Yulin Pan, Xiangteng He, Chaojie Mao, Zhen Han, Zeyinzi Jiang, Jingfeng Zhang, and Yu Liu. Icebench: A unified and comprehensive benchmark for image creating and editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

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 *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 22500–22510, 2023.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face. *Advances in Neural Information Processing Systems*, 36:38154–38180, 2023.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36, 2024.

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8228–8238, 2024.

Chenfei Wu, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv preprint arXiv:2303.04671*, 2023a.

Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023b.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023.

Siwei Yang, Mude Hui, Bingchen Zhao, Yuyin Zhou, Nataniel Ruiz, and Cihang Xie. Complex-edit: Cot-like instruction generation for complexity-controllable image editing benchmark. *arXiv preprint arXiv:2504.13143*, 2025.

Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing. In *Advances in Neural Information Processing Systems*, 2023a.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 3836–3847, 2023b.