STRIDE: SCALE-AWARE MULTI-AGENT EVOLUTIONARY REFINEMENT FOR INFOGRAPHIC GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Infographic generation challenges text-to-image models because a single image must satisfy numerous visual, textual, and semantic constraints. Iterative agentic refinement with a multimodal verifier is a natural remedy, but existing methods face a scale mismatch: local edits cannot correct global composition errors, while full regeneration may destroy already-correct regions. We introduce **STRIDE**, a constraint-grounded multi-agent framework that treats refinement over any frozen generator as a history-aware evolution strategy. STRIDE separates constraint evaluation—whether each requirement is satisfied—from visual grounding—where and how it is violated. It then routes global revisions to a composer subagent and local corrections to a retoucher subagent. Each subagent conducts internal planning and generates multiple candidates, while STRIDE accepts a candidate only when it improves the verifier score, reducing regression across iterations. Because stochastic generators and imperfect verifiers make any single step unreliable, it is this combination of scale-aware routing, per-step redundancy, and monotonic acceptance that yields stable improvement across iterations. With fully open-weight generator–verifier systems, STRIDE outperforms proprietary systems including Nano-Banana-2.0 and Seedream-5.0-Pro on BizGenEval, matches GPT-Image-2.0 and surpasses Seedream-5.0-Pro on IGenBench, and achieves a new state-of-the-art score on WISE, while substantially outperforming existing agentic methods across all evaluated benchmarks.

1 INTRODUCTION

Text-to-image (T2I) generation has made remarkable progress in producing visually compelling images, but generating reliable, information-rich visual content remains challenging (Li et al., 2026a; Tang et al., 2026). Infographics are a representative example (Xie et al., 2026), where text, numerical facts, charts, and spatial layout must work together to convey information correctly. A single incorrect value, malformed label, or inconsistent visual relationship can compromise the reliability of the entire infographic (Li et al., 2026b; Zhu et al., 2026a). However, existing generation paradigms primarily optimize image synthesis without explicitly verifying whether generated content satisfies its underlying constraints.

Recent models have made significant progress in infographic generation. Closed-source systems such as Nano-Banana 2.0 (Google DeepMind, 2026) and GPT-Image 2.0 (OpenAI, 2026b) exhibit strong visual generation capabilities, while open-source models such as ERNIE-Image (Liu et al., 2026) and Infographic-V2 (Diao et al., 2026) make infographic generation increasingly accessible and specialized. Beyond direct generation, recent agent-based approaches, including PaperBanana (Zhu et al., 2026a), AutoFigure (Zhu et al., 2026b), and PromptEnhancer (Wang et al., 2025), employ planning, feedback, and iterative refinement to further enhance generation quality. Nevertheless, existing approaches remain largely focused on visual plausibility and overall quality, with limited ability to explicitly verify fine-grained information constraints, localize their violations, and perform targeted correction. Distilling these limitations, we identify three key challenges in current agentic generation and refinement methods:

- **Insufficiently Grounded Feedback.** Holistic scores and coarse critiques provide limited guidance for targeted correction. Constraint-level judgments alone may still leave the location and nature of a failure unclear.

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.

- **Mismatch between Error and Edit Scales.** Infographic errors range from global composition failures to localized text defects. A single revision mechanism cannot address both effectively: local edits have limited structural reach, while full-image regeneration can disrupt correct content when only a small correction is needed.
- **Unreliable Progress across Iterations.** A revision may fix one requirement while violating another, and stochastic generation makes individual proposals unreliable.

To address these challenges, we introduce STRIDE, a training-free multi-agent framework for scale-aware evolutionary refinement over frozen text-to-image generators. STRIDE separates rubric-based verification from visual grounding, combining constraint-level judgments with spatial evidence to guide targeted correction. A trajectory-aware lead editor uses this feedback and prior attempts to route revisions to a composer for global regeneration or a retoucher for localized text and formula correction. Each selected sub-agent proposes multiple candidates, and a $(1 + \lambda)$-evolution strategy retains the current champion unless a candidate improves its verifier score, allowing exploration while limiting regression. STRIDE outperforms Nano-Banana-2.0 and Seedream-5.0-Pro on BizGenEval, matches GPT-Image-2.0 and surpasses Seedream-5.0-Pro on IGenBench, and achieves a new state-of-the-art score on WISE, while substantially outperforming existing agentic methods across all evaluated benchmarks. Our contributions are threefold:

- We introduce STRIDE, a training-free framework that formulates infographic generation as scale-aware evolutionary refinement over frozen generators, adapting the scope of correction to the errors in the current image.
- We develop a refinement loop that separates rubric verification from spatial grounding, uses refinement history to guide global or local corrections, and combines multiple proposals with verifier-guided champion retention to reduce regression.
- Experiments show that STRIDE outperforms contemporary agentic methods and selected proprietary systems, while demonstrating strong scalability across different generators and benchmarks. Extensive experiments characterize the contributions of grounding, execution routes, and champion evolution.

2 RELATED WORKS

Infographic Generation. Early text-to-image methods (Wu et al., 2025; Diao et al., 2026) progressively improved the rendering of accurate and legible text, and later systems (Gao et al., 2025; OpenAI, 2026b) extended this to multi-layer posters and commercial visuals. ChartGalaxy (Li et al., 2026b) and InfoChartQA (Xie et al., 2026) study chart understanding and chart-related visual reasoning, while BizGenEval (Li et al., 2026a) and IGenBench (Tang et al., 2026) evaluate whether generated visuals satisfy infographic-specified requirements. Proprietary models (Google DeepMind, 2025c; 2026; OpenAI, 2025; 2026b) have shown strong performance, but they still struggle to simultaneously satisfy a large number of discrete requirements and factual constraints in infographic generation. Open-weight models (Diao et al., 2026; Li et al., 2026a; Tang et al., 2026) fall further behind when faced with the dense constraints imposed by infographic requirements. STRIDE addresses this challenge through iterative dual-route evaluation and scale-aware refinement, enabling open-weight generators to match or even surpass proprietary models on this sophisticated task.

Agentic Augmentation. Agentic methods (Wang et al., 2025; He et al., 2026; Feng et al., 2026; Chen et al., 2026) enhance prompts by retrieving implicit details or reference images to condition generation, but their one-shot rewriting paradigm provides limited constraint enforcement and may overwrite content that is already correct. Methods such as Ag3 (Hahn et al., 2024) and AutoFigure (Zhu et al., 2026b) use editable graphs or SVGs to specify image layouts and conduct fixed-round layout-guided refinement. However, their reliance on proprietary models obscures whether gains come from the agentic framework or the base models, which also limits compatibility with most open-weight text-to-image generators. ReflectionFlow (Zhuo et al., 2025), T2I-Copilot (Chen et al., 2025), and PaperBanana (Zhu et al., 2026a) use verifier feedback to assess and refine generated outputs, but the absence of explicit rubric grounding makes failures under-specified, and their reliance on prompt revision or full-image regeneration can disrupt already correct regions and limit reliable improvement. Guided by dual-route evaluated failed rubrics and an evolution strat-

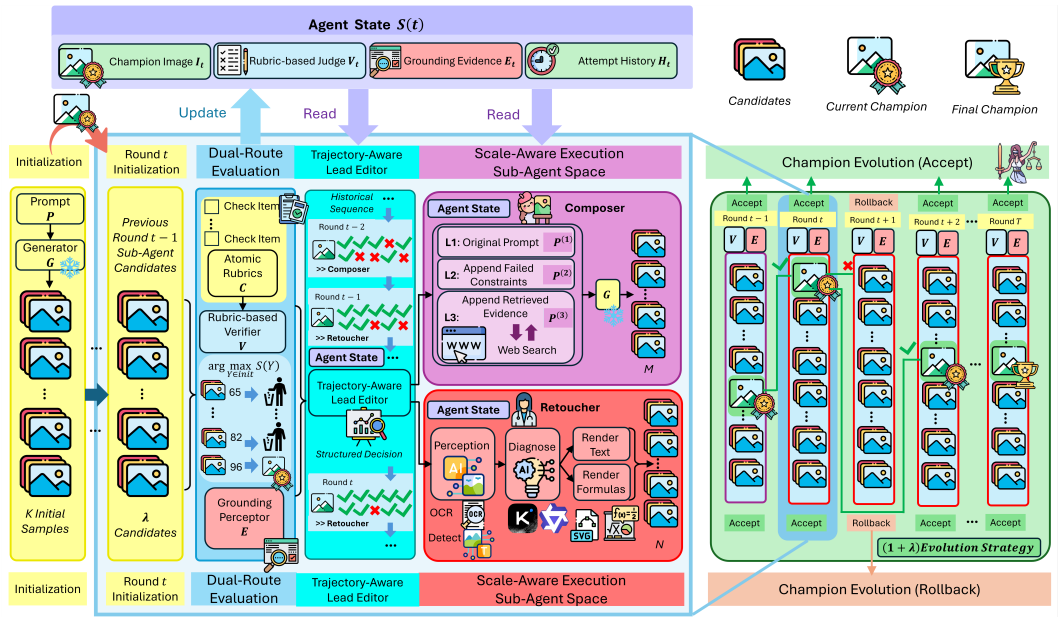

Figure 1: Overview of STRIDE. From K initial samples, STRIDE selects a champion and refines it for t rounds. Each round performs evolutionary refinement and agent state update through dual-route evaluation: the verifier scores candidate images against the rubrics, while the perceptor provides visual grounding of the observed failures. The lead editor then analyzes the agent state and refinement trajectory to route the champion to either the composer for global redraw or the retoucher for local editing. The generated candidates are then re-evaluated, and the champion is updated only when a candidate improves the verifier score. Repeating this process yields the final champion.

egy, STRIDE adaptively combines global regeneration with local editing to target specific errors, addressing the above limitations while ensuring stable improvement.

3 METHOD

3.1 OVERVIEW

As shown in Figure 1, STRIDE is a training-free multi-agent framework designed to iteratively refine images produced by any black-box text-to-image generator $G(P, z)$, where P is a prompt query and z a noise sample. We observe that generation failures often arise in two opposite regimes: global/compositional errors that local editing cannot repair, and localized residual errors that full re-generation would only perturb. STRIDE therefore operates over a scale-aware execution space with two complementary sub-agents – a **composer** that globally resamples the frozen generator and a **retoucher** that locally edits the current champion. The per-round choice between them is made by a central **lead editor**, which selects and guides the sub-agent based on the current agent state and the observations from a **dual-route evaluation** module. The core components of STRIDE are:

(1) **Dual-Route Evaluation.** STRIDE uses internal evaluations to drive iterative image refinement. Notably, it decouples evaluation into two complementary MLLM-based modules with distinct roles. The **verifier** performs closed-form judgment, scoring fine-grained semantic compliance against a rubric set. The **perceptor** performs open-form grounding, extracting region-level structured descriptions (object locations, rendered text, attribute–region bindings, etc.) that help localize where a failed rubric is violated. Separating judgment from grounding lets each module specialize, and supplies the editor with the spatial evidence a pass/fail score alone cannot provide.

(2) **Trajectory-Aware Lead Editor**. As the central controller, the lead editor maintains a running state of candidate history, prior discriminator observations and failure mode, and the detailed editing history. Rather than simple prompt rewriting, it performs reflective trajectory analysis, makes a

decision on which actor (composer or retoucher) to invoke, and then formulates prompt updates for the selected actor that target to address previous failures.

(3) **Scale-Aware Execution Sub-Agent Space**. To prevent generative drift during iterative image generation, STRIDE operates over a dual-branch action space defined by two sub-agents. The **composer** explores the global generative manifold by rewriting the prompt and then resampling the image generator G, while the **retoucher** identifies localized editing needs and invokes specialized computer-vision tools, guarded by the perceptor's spatial grounding. The output of each editing action is then admitted or rejected under an evolution strategy. The current champion enters a new round of image refinement, until the iteration budget is exhausted.

3.2 AGENT STATE

STRIDE orchestrates the image refinement process as an iterative loop. At each iteration t, the system maintains a state tuple S_t defined as:

$$S_t = \big(I_t,\ V_t,\ E_t,\ H_t\big). \tag{1}$$

where I_t denotes the current "champion" (the highest-scoring image admitted up to iteration t), $V_t = V(I_t)$ represents the rubric-based judgment from the verifier, E_t holds perceptor's grounding evidence (region-level localizations of where rubrics are violated) for I_t, and H_t logs the historical sequence of executed actions, candidate rejections and failure mode, and observation deltas before step t. The agent state is accessible to the lead editor, the composer, and the retoucher.

3.3 DUAL-ROUTE EVALUATION

At each iteration, candidates generated by the execution module are evaluated by a dual-discriminator measurement system. This module decouples image quality assessment into semantic rubric verification and structured perceptual evidence extraction.

Rubric-Based Verifier. The rubric-based verifier is a multimodal LLM (Appendix J.1) that inspects an image and answers questions according to a set of atomic natural-language rubrics $\mathcal{C} = \{c_1, c_2, \ldots, c_M\}$, where each rubric specifies a fine-grained requirement of image quality. Given a candidate image I, the verifier outputs a binary decision $v_c(I) \in \{0, 1\}$ for each constraint $c \in \mathcal{C}$. The overall satisfaction score is defined as:

$$r(I; \mathcal{C}) = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} v_c(I). \tag{2}$$

In the case of handling hierarchical or prioritized constraints, we can further partition $\mathcal{C}$ into L ordered priority tiers $\big(\mathcal{C}^{(1)}, \ldots, \mathcal{C}^{(L)}\big)$. An image is scored via a multi-dimensional vector:

$$Score(I) = \Big(r(I; \mathcal{C}^{(1)}), \ldots, r(I; \mathcal{C}^{(L)})\Big), \tag{3}$$

where candidate vectors are evaluated via lexicographical ordering ($\succ$). For benchmarks with predefined checklists (e.g., BizGenEval, IGenBench), $\mathcal{C}$ is directly mapped from the evaluation checklists. For benchmarks that lack a detailed quality requirement checklist (e.g., WISE), rubrics can be dynamically constructed via a prompt parser $\mathcal{C} = \Phi(P)$, where Φ generates image-answerable constraints from the prompt (Appendix A).

Visual Grounding via the Perceptor. While the verifier issues rubric-level pass/fail judgments, the perceptor grounds those judgments spatially: it localizes objects, reads rendered text, and reports region-level attributes (Appendix J.2). This grounding resolves where and how a constraint fails, which the retoucher needs to target its local edits. The perceptor is applied to every round candidate alongside the verifier, but champion selection is driven by the verifier score S alone. After each round, only the grounding of the admitted champion is written into the state as $E_t = E(I_t)$. It's also worth noting that we tested using specialized detection/OCR tools as perceptor and they underperformed a strong MLLM model like Kimi-K3 (Appendix F.2).

3.4 TRAJECTORY-AWARE LEAD EDITOR

A single editing mechanism is often insufficient for iterative image refinement. When the current image contains global semantic or compositional failures, local editing is too weak to recover the

missing structure. Conversely, when only a small subset of attributes remains incorrect, full-image regeneration introduces unnecessary drift. To balance these two regimes, STRIDE operates over a scale-aware action space with two complementary operators: global redraw, executed by the composer, and local refinement, executed by the retoucher.

The lead editor serves as the central controller of STRIDE. At round t, it reads the current agent state $S_t = \big(I_t,\ V_t,\ E_t,\ H_t\big)$ and performs reflective trajectory analysis. It then determines which editing sub-agent to invoke next and what editing guidance G_t to pass to it (Appendix J.3). The lead editor invokes the composer for global failure modes, including missing objects, incorrect layouts, and compositional errors that jointly violate multiple constraints. It invokes the retoucher when the residual errors are localized and spatially addressable, such as text-rendering errors, minor attribute mismatches, or confined visual defects.

3.5 SCALE-AWARE EXECUTION SUB-AGENT SPACE

Global Redraw via the Composer. The composer produces a fresh full-image hypothesis by resampling the frozen generator under failure-aware prompt conditioning (Appendix J.4). Rather than rewriting the source prompt directly, it preserves original prompt P as a prefix and appends an instruction block derived from the currently failed constraints, yielding three prompt-conditioning levels:

$$P^{(\ell)} = \begin{cases} P, & \ell = 1, \\ P \oplus A(P, \mathcal{F}_t, G_t), & \ell = 2, \\ P \oplus A\big(P, \mathcal{F}_t, G_t, R(\mathcal{F}_t)\big), & \ell = 3, \end{cases} \tag{4}$$

where $\oplus$ appends an instruction block while preserving P, $A(\cdot)$ is produced by prompting an LLM to convert failed constraints $\mathcal{F}_t \in V_t$ into imperative guidance for the generator, while conditioning on the lead editor's guidance G_t. When the composer judges that an external web search tool is needed, the retrieved evidence $R(\cdot)$ is also included in the prompt.

The composer autonomously selects between level $\ell = 2$ and $\ell = 3$. It first reasons internally from the source prompt, the failed constraints, the verifier rationales, together with the guidance generated by the lead editor, then emitting a candidate instruction block together with a self-assessed confidence τ_c over the required knowledge those constraints depend on (names, dates, quantities), not over its own writing. If the confidence exceeds a threshold $\tau_c > \tau_c^*$, the composer adopts the instruction block ($\ell = 2$). Otherwise, it issues self-generated queries, retrieves snippets, and revises the block against the retrieved evidence via another LLM call ($\ell = 3$). Level ($\ell = 1$) is a fallback outside the model call: when no valid instruction block is produced, the framework redraws from the original prompt P unchanged. Given the selected planning level ℓ, the composer samples the generator G to form a candidate set

$$Y_t^{\text{Composer}} = \{G(P^{(\ell)}, z_m)\}_{m=1}^{M}, \tag{5}$$

where $z_1, \ldots, z_M$ are independent noise draws and M is the proposal budget for the composer.

Local Surgery via the Retoucher. The retoucher improves the current champion through targeted local edits (Appendix J.5) rather than regenerating the entire image. It first queries the perceptor for region-level groundings $B(I_t)$: indexed OCR text blocks that localize content, each paired with the checklist questions it fails. Because perception is stochastic, a single pass may miss or mislocate an erroneous region. To reduce this risk, the retoucher therefore draws N parallel branches, each targeting a single region from an independent sample of $B(I_t)$.

Branches may address the same or different regions, so parallel sampling diversifies both localization and the proposed corrections. In branch n, the retoucher reasons from the source prompt, failed constraints, verifier rationales, lead editor's guidance and $B(I_t)$, emitting a single edit instruction T_n with confidence τ_r. The instruction T_n binds one block index to a corrected string, rendered in LaTeX for formulas and as plain text otherwise. The edit is then applied to the champion, which can be formulated as:

$$\tilde{I}_{t,n} = \text{RenderEdit}(I_t, B(I_t), T_n). \tag{6}$$

We filter branches before rendering. Branch n is discarded if $\tau_r < \tau_r^*$ or if its edit fingerprint (i.e. the block index and its replacement) duplicates another branch, resulting $N' \leq N$ branches.

Each surviving branch is realized by the SVG toolbox, which implements $\mathrm{RenderEdit}$ locally: it masks the selected box with the sampled background color, renders the correction as a transparent PNG, and recovers the original typographic attributes so that the edit aligns seamlessly while the surrounding layout stays fixed. The surviving branches form the candidate set

$$Y_t^{\mathrm{Retoucher}} = \{\tilde{I}_{t,n}\}_{n=1}^{N'}, \qquad N' \le N. \tag{7}$$

When an attempt is rejected, the champion is left unchanged. The grounding is re-sampled next round, so localization keeps varying, while the lead editor's guidance accumulates across attempts—carrying forward corrections and regression warnings (e.g. an edit that repaired one block but broke a correct one) that steer later branches away from repeated mistakes.

3.6 Champion Evolution

Candidates from the composer and the retoucher are placed in a shared execution space. The same measurement module scores all candidates, and the resulting verifier scores determine the champion under an evolution strategy. At round t, the active sub-agent (i.e. composer or retoucher) generates a candidate set Y_t. We first identify the top candidate $Y_t^\star \in Y_t$ via:

$$Y_t^\star = \arg\max_{Y \in Y_t} Score(Y). \tag{8}$$

The champion image is updated via an $(1+\lambda)$**-evolution strategy** to avoid image quality drift:

$$I_{t+1} = \begin{cases} Y_t^\star, & \text{if } Score(Y_t^\star) \succ Score(I_t), \\ I_t, & \text{otherwise.} \end{cases} \tag{9}$$

where 1 denotes current champion, λ denotes the number of new champion candidates generated by the active sub-agent, $\succ$ is the lexicographical ordering. The state I_t updates strictly upon achieving a superior satisfaction vector, with ties preserving the incumbent champion. The initial champion I_0 is selected similarly from K samples generated at $t = 0$. Each round consumes one unit of the round budget (independent of the M redraw samples or N retouch branches produced within it), appends unselected candidates to H_t, and halts upon reaching full rubric satisfaction, budget exhaustion, or execution failures. Then the lead editor starts a new round of refinement.

4 Experiments

4.1 Benchmarks and Setup

Benchmarks. To fairly compare STRIDE with contemporary proprietary and open-source T2I models as well as agentic generation methods, we evaluate its infographic generation and refinement capability on two large-scale public benchmarks: *(1) BizGenEval* (Li et al., 2026a), *(2) IGen-Bench* (Tang et al., 2026). To evaluate the scalability of STRIDE, we further conduct general-purpose, non-infographic-specific experiments on the *(3) WISE* (Niu et al., 2025) dataset.

Setup. STRIDE uses Kimi-K3 (Kimi et al., 2026a) as the lead editor, perceptor, composer, and retoucher, with open-weight text-to-image generators including ERNIE-Image (Liu et al., 2026), Infographic-V2 (Diao et al., 2026), and Qwen-Image-2512 (Wu et al., 2025). The verifier is the open-weight Qwen3.5-397B-A17B (Qwen Team, 2026b) model. The agent runs for at most $t = 3$ iterations under a $(1+\lambda)$-evolution strategy with $\lambda = 4$; the initial champion I_0 is selected from the best-of-$K = 6$ samples at $t = 0$. The composer samples $M = 4$ candidates from the frozen generator, and the retoucher launches $N = 4$ parallel local editing branches. We set the internal confidence thresholds to $\tau_c^* = 0.7$ for the composer and $\tau_r^* = 0.8$ for the retoucher. Final benchmark evaluation uses the corresponding official evaluator for each benchmark (Appendix B).

4.2 Main Results

BizGenEval. Table 1 shows that STRIDE substantially improves both open-weight generators, outperforms existing agentic baselines and proprietary systems including Nano-Banana-2.0 (Google DeepMind, 2026) and Seedream-5.0-Pro (ByteDance, 2026). With ERNIE-Image and Infographic-V2, STRIDE achieves overall Hard scores of 51.1 and 69.5, improving over the corresponding frozen

Table 1: Quantitative comparison on BizGenEval. Each entry reports the hard score. The proprietary-model results are reproduced unchanged from the official BizGenEval.

Model	Venue	Params	Score				Overall
			Layout	Attribute	Text	Knowledge	Average
Proprietary models							
Nano-Banana-Pro	25@Google	–	72.2	65.6	86.4	82.6	76.7
Wan2.6-T2I	26@Alibaba	–	46.4	16.6	12.6	12.2	21.9
Seedream-5.0	26@ByteDance	–	67.6	42.4	43.4	41.8	48.8
Seedream-5.0-Pro	26@ByteDance	–	75.5	74.1	63.2	49.8	65.7
GPT-Image-1.5	26@OpenAI	–	51.6	25.8	40.4	26.0	35.9
Nano-Banana-2.0	26@Google	–	68.4	57.4	83.4	64.6	68.5
Open-weight models							
Qwen-Image-2512	25@Alibaba	20B	22.2	1.2	1.8	0.0	6.3
Z-Image	25@Z-Image	6B	26.8	2.6	2.8	0.6	8.2
HunyuanImage-3.0	25@Tencent	80B-A13B	27.8	13.8	10.2	0.0	13.0
ERNIE-Image	26@Baidu	8B	51.8	31.8	28.3	0.1	28.0
Infographic-V2	26@SenseTime	8B	58.8	42.6	63.0	0.0	41.1
Agentic framework							
PE (ERNIE-Image)	26@Baidu	8B-3B	39.2	27.4	26.0	4.2	24.2
GenEvolve (ERNIE-Image)	26@Arxiv	8B	42.8	27.0	18.8	2.8	22.9
GenEvolve (ERNIE-Image-PE)	26@Arxiv	8B-3B	35.6	20.4	19.4	6.2	20.4
Gen-Searcher (ERNIE-Image)	26@Arxiv	8B	47.6	30.2	23.6	9.4	27.7
Gen-Searcher (Infographic-V2)	26@Arxiv	8B	58.2	38.6	58.2	17.8	43.2
PromptEnhancer-7B (ERNIE-Image)	26@CVPR	8B	32.4	9.0	6.2	0.4	12.0
PromptEnhancer-32B (ERNIE-Image)	26@CVPR	8B	35.4	12.6	8.0	1.2	14.3
PromptEnhancer-7B (Infographic-V2)	26@CVPR	8B	30.6	5.8	2.2	0.6	9.8
PromptEnhancer-32B (Infographic-V2)	26@CVPR	8B	36.0	10.0	8.6	1.4	14.0
STRIDE (ERNIE-Image)	**Ours**	8B	64.4	50.8	55.2	34.0	51.1
STRIDE (Infographic-V2)	**Ours**	8B	**79.2**	**65.8**	**79.2**	**54.0**	**69.5**

Table 2: Quantitative comparison on IGenBench. The proprietary-model results are reproduced unchanged from the official IGenBench.

Model	Venue	Params	Question Type Q-ACC										Overall	
			Comp.	Enc.	Order	Marks	Anno.	Axes	Leg.	Chart	Title	Deco.	Q-ACC	I-ACC
Proprietary models														
Image-01	25@MiniMax	–	0.01	0.05	0.04	0.10	0.10	0.14	0.03	0.22	0.14	0.47	0.13	0.00
P-Image	25@PrunaAI	–	0.08	0.15	0.19	0.27	0.36	0.28	0.54	0.43	0.58	0.68	0.34	0.00
Nano-Banana	25@Google	–	0.18	0.31	0.27	0.44	0.54	0.57	0.52	0.60	0.65	0.81	0.48	0.02
Nano-Banana-Pro	25@Google	–	0.84	0.86	0.90	0.87	0.93	0.93	0.96	0.92	0.98	0.94	0.90	0.49
GPT-Image-1.5	26@OpenAI	–	0.38	0.48	0.44	0.57	0.50	0.54	0.57	0.68	0.60	0.80	0.55	0.12
Seedream-4.5	26@ByteDance	–	0.34	0.37	0.47	0.48	0.70	0.70	0.81	0.68	0.95	0.84	0.61	0.06
Seedream-5.0-Pro	26@ByteDance	–	0.64	0.47	0.71	0.73	0.77	0.85	0.83	0.83	0.86	0.88	0.73	0.17
GPT-Image-2.0	26@OpenAI	–	0.89	0.61	0.93	0.87	0.94	0.98	0.95	0.93	0.99	0.95	0.88	0.38
Qwen-Image-3.0	26@Alibaba	–	0.65	0.43	0.77	0.77	0.84	0.84	0.87	0.85	0.80	0.88	0.74	0.13
Open-weight models														
Qwen-Image	25@Alibaba	20B	0.10	0.13	0.19	0.29	0.43	0.37	0.51	0.48	0.56	0.78	0.36	0.01
SenseNova-U1	26@SenseTime	8B-A3B	0.17	0.22	0.33	0.41	0.36	0.49	0.56	0.60	0.55	0.78	0.42	0.02
ERNIE-Image	26@Baidu	8B	0.17	0.16	0.25	0.34	0.43	0.47	0.61	0.55	0.57	0.81	0.41	0.01
Infographic-V2	26@SenseTime	8B	0.54	0.49	0.69	0.66	0.79	0.80	0.85	0.80	0.94	0.88	0.72	0.17
Agentic framework														
Gen-Searcher (ERNIE-Image)	26@Arxiv	8B	0.17	0.12	0.30	0.39	0.44	0.46	0.53	0.58	0.50	0.77	0.40	0.01
Gen-Searcher (Infographic-V2)	26@Arxiv	8B	0.39	0.29	0.54	0.51	0.73	0.69	0.78	0.68	0.90	0.84	0.60	0.07
PromptEnhancer-32B (ERNIE-Image)	26@CVPR	8B	0.21	0.24	0.27	0.39	0.47	0.47	0.70	0.60	0.59	0.80	0.45	0.01
PromptEnhancer-32B (Infographic-V2)	26@CVPR	8B	0.29	0.17	0.32	0.46	0.55	0.56	0.80	0.60	0.89	0.80	0.52	0.05
STRIDE (ERNIE-Image)	**Ours**	8B	0.30	0.32	0.41	0.52	0.60	0.63	0.73	0.70	0.67	0.89	0.55	0.03
STRIDE (Infographic-V2)	**Ours**	8B	**0.70**	**0.72**	**0.81**	**0.78**	**0.90**	**0.89**	**0.92**	**0.91**	**0.98**	**0.94**	**0.84**	**0.38**

generators by 23.1 and 28.4 points, respectively. It also exceeds the strongest agentic baseline–Gen-Searcher (Feng et al., 2026) on the two generators by 23.4 and 26.3 points. These gains are consistent across all four capabilities: for Infographic-V2, layout, attribute, text, and knowledge scores improve by 20.4, 23.2, 16.2, and 54.0 points, respectively, indicating that STRIDE enhances factual and constraint-level correctness rather than merely improving visual layout. Notably, using only open-weight components, STRIDE (Infographic-V2) attains a score of 69.5, surpassing the proprietary Nano-Banana-2.0 (68.5) and Seedream-5.0-Pro (65.7), demonstrating strong competitiveness.

IGenBench. As presented in Table 2, using Infographic-V2, STRIDE achieves 0.84 Q-ACC and 0.38 I-ACC, compared with 0.72 and 0.17 for the frozen generator and 0.60 and 0.07 for the strongest agentic baseline, Gen-Searcher (Feng et al., 2026). STRIDE (ERNIE-Image) also improves over its base generator, increasing Q-ACC from 0.41 to 0.55. With all open-weight components, STRIDE (Infographic-V2) attains an I-ACC of 0.38, matching GPT-Image-2.0 (OpenAI, 2026b). Moreover, its Q-ACC and I-ACC both surpass those of advanced proprietary systems, including

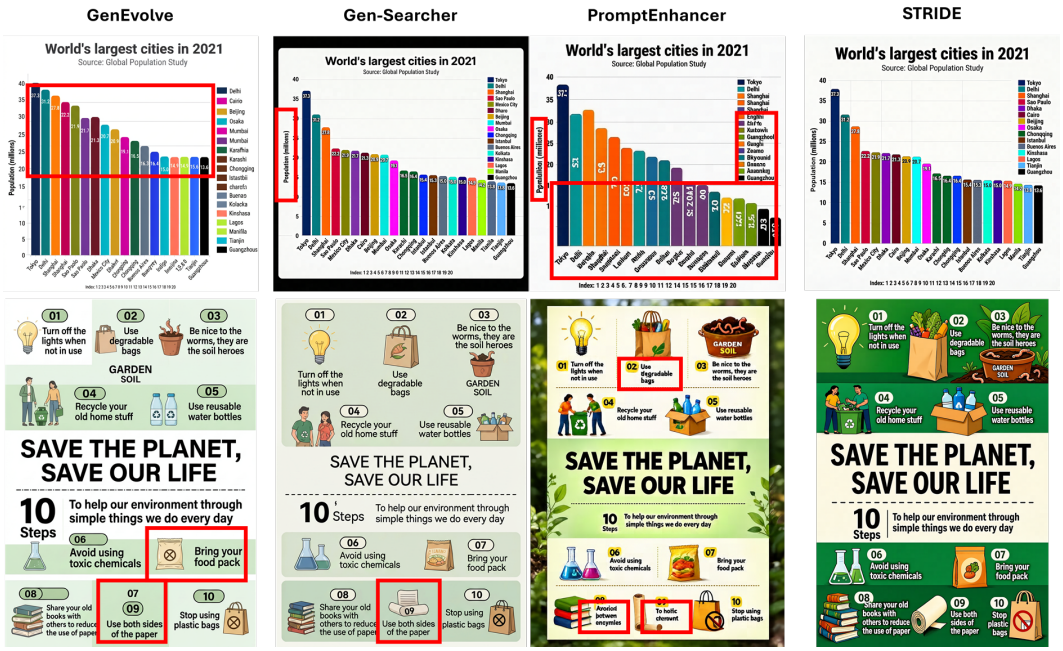

Figure 2: Qualitative comparison with existing agentic methods. Red boxes highlight errors.

Table 3: STRIDE ablation study with ERNIE-Image as the base generator. We compare sampling-only baselines, execution routes, perception grounding, and refinement iterations on BizGenEval.

#	Components					BizGenEval									**Avg.**
	Lead	Comp.	Ret.	Perc.	Iters.	Slides	Webpage	Poster	Chart	Sci. Fig.	Layout	Attr.	Text	Know.	**Score**
(1)		Best-of-1			1 iter	32.8	41.5	35.6	10.5	19.7	51.8	31.8	28.3	0.1	28.0
(2)		Best-of-6			1 iter	39.0	48.5	40.2	19.0	26.8	60.6	41.4	36.4	0.4	34.7
(3)	✓	✓	–	–	3 iters	48.5	53.2	44.8	21.2	38.0	63.8	45.4	40.6	14.8	41.1
(4)	✓	–	✓	–	3 iters	57.5	60.5	52.0	38.8	42.5	64.4	41.4	60.4	34.8	50.2
(5)	✓	✓	✓	–	3 iters	57.0	57.0	54.5	34.8	41.5	64.2	49.4	51.0	31.2	48.9
(6)	✓	–	✓	✓	3 iters	58.2	61.8	54.2	36.5	41.5	63.6	44.0	59.0	35.2	50.4
(7)	✓	✓	✓	✓	1 iter	54.2	56.5	51.2	30.8	39.0	63.2	45.8	51.6	24.8	46.4
(8)	✓	✓	✓	✓	2 iters	56.0	59.0	51.5	34.0	40.2	63.6	47.8	53.8	27.4	48.1
(9)	✓	✓	✓	✓	3 iters	61.3	60.2	54.5	37.0	42.5	64.4	50.8	55.2	34.0	51.1

Nano-Banana (Google DeepMind, 2025a), Seedream-5.0-Pro (ByteDance, 2026), and Qwen-Image-3.0 (Qwen Team, 2026a).

Qualitative Results. As shown in Figure 2, compared with contemporary advanced agentic frameworks, STRIDE produces infographics that better satisfy the specified constraints, exhibit more coherent layouts, place semantic attributes more appropriately, and render text more accurately. These results meet the stringent requirements of infographic generation and demonstrate clear advantages. Additional qualitative results and a detailed case study are provided in Appendices G and H.

4.3 ABLATION STUDIES

Table 3 ablates the main components of STRIDE using ERNIE-Image as the base generator. We report BizGenEval hard scores. Rows (1)–(2) compare against sampling-only baselines, while the remaining rows progressively enable the lead editor, composer, retoucher, perceptor, and multiple refinement iterations. Detailed results are provided in Appendix D.2.

Candidate Exploration. Increasing the candidate pool from Best-of-1 to Best-of-6 (Table 3 (1)-(2)) raises the BizGenEval average from 28.0 to 34.7, but the result remains far below the full system's

Table 4: Comparison of champion-evolution strategies on BizGenEval. The $(1, 1)$ and $(1, 4)$ variants denote forced replacement, whereas the $(1 + 1)$ and $(1 + 4)$ variants retain the incumbent champion unless a better candidate is found.

Strategy	Capability				Average
	Layout	Attr.	Text	Know.	
Random selection	54.6	40.0	46.4	20.6	40.4
$(1, 1)$ forced replacement	56.0	37.8	42.2	21.4	39.4
$(1 + 1)$-evolution strategy	62.6	43.6	49.2	26.8	45.6
$(1, 4)$ forced replacement	59.2	42.4	50.4	29.6	45.4
$(1 + 4)$-**evolution strategy (ours)**	64.4	50.8	55.2	34.0	**51.1**

Table 5: Human and automated evaluation of the paired subset from BizGenEval and IGenBench. The frozen generator and STRIDE outputs are evaluated using human consensus (H), the Qwen3.5-397B-A17B verifier (Q), and the corresponding official benchmark evaluator (O).

Benchmark	Generator	Frozen generator			STRIDE			Δ (STRIDE − Frozen)			Correlation	
		H	Q	O	H	Q	O	H	Q	O	$\rho_{\mathrm{H-Q}}$	$\rho_{\mathrm{H-O}}$
BizGenEval: Hard Score												
BizGenEval	ERNIE-Image	30.1	39.0	32.8	50.4	63.8	53.8	+20.3	+24.8	+21.0	0.650	0.732
BizGenEval	Infographic-V2	41.5	40.0	42.4	71.3	77.6	70.6	+29.8	+37.6	+28.2	0.725	0.811
IGenBench: Q-ACC												
IGenBench	ERNIE-Image	0.42	0.50	0.43	0.58	0.66	0.57	+0.16	+0.16	+0.14	0.708	0.733
IGenBench	Infographic-V2	0.76	0.82	0.77	0.88	0.93	0.89	+0.12	+0.11	+0.12	0.785	0.847

score of 51.1. Notably, the knowledge score stays close to zero across all candidate-pool sizes. Thus, the gains require structured diagnosis and targeted refinement, rather than mere best-of-K sampling.

Scale-Aware Execution. Composer-only (Table 3 (3):(2)) improves the average to 41.1, with a large gain on knowledge constraints over the base model. Retoucher-only (Table 3 (4):(2)) is stronger, reaching 50.2 and substantially improving text score from 36.4 to 60.4. The complete scale-aware system (Table 3 (9):(2)) achieves the best score of 51.1, which supports the scale-aware design.

Dual-Route Evaluation. In particular, adding the perceptor to the composer–retoucher configuration (Table 3 (9):(5)) improves the three-iteration score from 48.9 to 51.1, suggesting that grounding is useful when it supports route selection and local editing jointly.

Iterative Refinement. With all components enabled, performance increases from 46.4 to 48.1 and 51.1 across one, two, and three iterations (Table 3 (7)-(8)-(9)), showing that the refinement loop accumulates improvements rather than merely adding stochastic variation.

Evolution Strategy. Table 4 shows that both broader candidate exploration and incumbent retention contribute to our $(1 + \lambda)$-evolution strategy. Increasing the candidate set (λ) from one to four raises the average score from 39.4 to 45.4 under forced replacement, and from 45.6 to 51.1 when retaining the incumbent, indicating that a larger candidate pool improves the chance of finding a strong proposal. Moreover, at the same candidate budget, incumbent retention yields gains of 6.2 points for $(1+1)$ over $(1, 1)$ and 5.7 points for $(1+4)$ over $(1, 4)$, as it avoids discarding the current champion for a verifier-non-improving candidate. More detailed analysis in Appendix F.3.

4.4 HUMAN-MLLM STUDIES.

We examine whether automated evaluation aligns with human judgments and whether the gains of STRIDE persist under human evaluation (Appendix E). Table 5 shows positive correlations with human scores for both the Qwen verifier ($\rho = 0.650$–0.785) and the official evaluators ($\rho = 0.732$–0.847). Human evaluation also preserves the improvement of STRIDE in all four configurations: BizGenEval hard score increases by 20.3 and 29.8 points for ERNIE-Image and Infographic-V2, respectively, while IGenBench Q-ACC increases by 0.16 and 0.12. These results support human alignment of the automated rankings and provide independent evidence for the refinement gains.

5 CONCLUSION

We presented STRIDE, a training-free multi-agent framework that formulates infographic generation as scale-aware evolutionary refinement over frozen text-to-image generators. STRIDE separates rubric-based verification from spatial grounding, routes failures to either global composition or local retouching according to their scale, and admits only candidates that improve verifier-scored constraint satisfaction under an evolution strategy. Across BizGenEval, IGenBench and WISE, STRIDE consistently improves open-weight generators, outperforms prior agentic enhancement methods, and matches or surpasses proprietary systems. We hope this work inspires future advancements in agentic infographic generation and refinement.

AI USE STATEMENT

Large Language Models (LLMs) were used in this study to aid in polishing the manuscript. Specifically, we used LLMs to assist in refining the language and detecting potential grammatical errors. This is to improve readability and ensure clarity of the paper. We confirm that LLMs were not involved in research ideation, method exploration, and experiment designs. All research ideas, methods, and analysis were produced by the authors. We take full responsibility for the content in this paper, including the text generated or polished by the LLMs.

REFERENCES

ByteDance. Seedream 5.0 Pro. https://seed.bytedance.com/en/seedream5_0_pro, 2026. Accessed: 2026-09-18.

Chieh-Yun Chen, Min Shi, Gong Zhang, and Humphrey Shi. T2i-copilot: A training-free multi-agent text-to-image system for enhanced prompt interpretation and interactive generation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19396–19405. IEEE, 2025.

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*, 2026.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Cheng Cui, Yubo Zhang, Ting Sun, Xueqing Wang, Hongen Liu, Manhui Lin, Yue Zhang, Tingquan Gao, Changda Zhou, Jiaxuan Liu, et al. Pp-ocrv5: A specialized 5m-parameter model rivaling billion-parameter vision-language models on ocr tasks. *arXiv preprint arXiv:2603.24373*, 2026.

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.

Kaituo Feng, Manyuan Zhang, Shuang Chen, Yunlong Lin, Kaixuan Fan, Yilei Jiang, Hongyu Li, Dian Zheng, Chenyang Wang, and Xiangyu Yue. Gen-searcher: Reinforcing agentic search for image generation. *arXiv preprint arXiv:2603.28767*, 2026.

Yifan Gao, Zihang Lin, Chuanbin Liu, Min Zhou, Tiezheng Ge, Bo Zheng, and Hongtao Xie. Postermaker: Towards high-quality product poster generation with accurate text rendering. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8083–8093. IEEE, 2025.

Google DeepMind. 2.5 flash model card. https://storage.googleapis.com/deepmind-media/Model-Cards/Gemini-2-5-Flash-Model-Card.pdf, 2025a. Accessed: 2026-09-19.

Google DeepMind. Gemini 3 flash: Best for frontier intelligence at speed. https://deepmind.google/models/gemini/flash/, 2025b. Accessed: 2026-09-18.

Google DeepMind. Gemini 3 pro image model card. https://storage.googleapis.com/deepmind-media/Model-Cards/Gemini-3-Pro-Image-Model-Card.pdf, 2025c. Accessed: 2026-09-19.

Google DeepMind. Nano Banana 2: Combining Pro capabilities with lightning-fast speed. https://blog.google/innovation-and-ai/technology/ai/nano-banana-2/, February 2026. Accessed: 2026-09-18.

Meera Hahn, Wenjun Zeng, Nithish Kannen, Rich Galt, Kartikeya Badola, Been Kim, and Zi Wang. Proactive agents for multi-turn text-to-image generation under uncertainty. *arXiv preprint arXiv:2412.06771*, 2024.

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.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Team Kimi, Tongtong Bai, Yifan Bai, Yiping Bao, Jianfeng Cai, Xinyuan Cai, Peizhou Cao, Yuxuan Cao, Ziwei Chai, Y Charles, et al. Kimi k3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026a.

Team Kimi, Tongtong Bai, Yifan Bai, Yiping Bao, SH Cai, Yuan Cao, Ziwei Chai, Y Charles, HS Che, Cheng Chen, et al. Kimi k2. 5: Visual agentic intelligence. *arXiv preprint arXiv:2602.02276*, 2026b.

Yan Li, Zezi Zeng, Ziwei Zhou, Xin Gao, Muzhao Tian, Yifan Yang, Mingxi Cheng, Qi Dai, Yuqing Yang, Lili Qiu, et al. Bizgeneval: A systematic benchmark for commercial visual content generation. *arXiv preprint arXiv:2603.25732*, 2026a.

Zhen Li, Duan Li, Yukai Guo, Xinyuan Guo, Bowen Li, Lanxi Xiao, Shenyu Qiao, Jiashu Chen, Zijian Wu, Hui Zhang, et al. Chartgalaxy: A dataset for infographic chart understanding and generation. In *International Conference on Learning Representations*, volume 2026, pp. 52484–52546, 2026b.

Jiaxiang Liu, Zhida Feng, Pengyu Zou, Zhenyu Qian, Tianrui Zhu, Jun Xia, Yuehu Dong, Yanzheng Lin, Honglin Xiong, Anqi Chen, et al. Ernie-image technical report. *arXiv preprint arXiv:2605.25347*, 2026.

Moonshot AI. Kimi K2.7 Code: Open-source-agent-modell für coding. https://www.kimi.ai/de/resources/kimi-k2-7-code, 2026. Accessed: 2026-09-18.

Yuwei Niu, Munan Ning, Mengren Zheng, Weiyang Jin, Bin Lin, Peng Jin, Jiaqi Liao, Chaoran Feng, Fanqing Meng, Kunpeng Ning, et al. Wise: A world knowledge-informed semantic evaluation for text-to-image generation. *arXiv preprint arXiv:2503.07265*, 2025.

OpenAI. The new ChatGPT Images is here. https://openai.com/index/new-chatgpt-images-is-here/, December 2025. Accessed: 2026-09-18.

OpenAI. GPT-5.5 System Card. https://deploymentsafety.openai.com/gpt-5-5, 2026a. Accessed: 2026-09-18.

OpenAI. Introducing ChatGPT Images 2.0. https://openai.com/index/introducing-chatgpt-images-2-0/, 2026b. Accessed: 2026-09-18.

Qwen Team. Qwen-image-3.0: Rich content, authentic details, deep knowledge. https://qwen.ai/blog?id=qwen-image-3.0, August 2026a. Accessed: 2026-09-18.

Qwen Team. Qwen3.5: Towards Native Multimodal Agents. https://qwenlm.github.io/blog/qwen3.5/, February 2026b. Accessed: 2026-09-18.

Qwen Team. Qwen3.7-Plus: Multimodal Agent Intelligence. https://qwenlm.github.io/blog/qwen3.7-plus/, May 2026c. Accessed: 2026-09-18.

Yinghao Tang, Xueding Liu, Boyuan Zhang, Tingfeng Lan, Yupeng Xie, Jiale Lao, Yiyao Wang, Haoxuan Li, Tingting Gao, Bo Pan, et al. Igenbench: Benchmarking the reliability of text-to-infographic generation. *arXiv preprint arXiv:2601.04498*, 2026.

Linqing Wang, Ximing Xing, Yiji Cheng, Zhiyuan Zhao, Donghao Li, Tiankai Hang, Jiale Tao, Qixun Wang, Ruihuang Li, Comi Chen, et al. Promptenhancer: A simple approach to enhance text-to-image models via chain-of-thought prompt rewriting. *arXiv preprint arXiv:2509.04545*, 2025.

Shihao Wang, Shilong Liu, Yuanguo Kuang, Xinyu Wei, Yangzhou Liu, Zhiqi Li, Yunze Man, Guo Chen, Andrew Tao, Guilin Liu, et al. Locateanything: Fast and high-quality vision-language grounding with parallel box decoding. In *European Conference on Computer Vision*, pp. 336–357. Springer, 2026.

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.

Tianchi Xie, Minzhi Lin, Mengchen Liu, Yilin Ye, Changjian Chen, and Shixia Liu. Infochartqa: A benchmark for multimodal question answering on infographic charts. *Advances in Neural Information Processing Systems*, 38, 2026.

Yan Zhou. A yolo-nl object detector for real-time detection. *Expert Systems with Applications*, 238: 122256, 2024.

Dawei Zhu, Rui Meng, Yale Song, Xiyu Wei, Sujian Li, Tomas Pfister, and Jinsung Yoon. Paperbanana: Automating academic illustration for ai scientists. In *Forty-third International Conference on Machine Learning*, 2026a.

Minjun Zhu, Zhen Lin, Yixuan Weng, Panzhong Lu, Qiujie Xie, Yifan Wei, Sifan Liu, Qiyao Sun, and Yue Zhang. Autofigure: Generating and refining publication-ready scientific illustrations. *arXiv preprint arXiv:2602.03828*, 2026b.

Le Zhuo, Liangbing Zhao, Sayak Paul, Yue Liao, Renrui Zhang, Yi Xin, Peng Gao, Mohamed Elhoseiny, and Hongsheng Li. From reflection to perfection: Scaling inference-time optimization for text-to-image diffusion models via reflection tuning. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15329–15339. IEEE, 2025.

APPENDIX CONTENTS

A Rubric Checklist Instantiation **14**

B Benchmark Details **15**

C Advantages over Existing Agentic Methods **16**

D Extended Experiments **17**

D.1 Extended Main Results 17

D.2 Extended Ablation Results 18

D.3 Extended Oracle Experiment Results 19

D.4 Extended Model Scalability Results on WISE 20

E Human-MLLM Studies Setup **21**

F Diagnostic Experiments **22**

F.1 Choice of Rubric-Based Verifier 22

F.2 Choice of Perceptor 22

F.3 Choice of Evolution Strategy 24

G More Qualitative Results **25**

H Case Study of Stride **29**

I Limitations and Future Work **31**

J Prompts for Agents in Stride **31**

J.1 Verifier 31

J.2 Perceptor 32

J.3 Lead Editor 32

J.4 Composer 33

J.5 Retoucher 35

A RUBRIC CHECKLIST INSTANTIATION

STRIDE requires a set of atomic rubrics that can be evaluated on an image. This appendix describes how that set is obtained on each benchmark. On BizGenEval and IGenBench we take it from the released checklist. On WISE, which provides no checklist, STRIDE constructs it dynamically from the prompt with the prompt parser Φ.

BizGenEval. Each prompt is paired with 20 verification questions, 10 easy and 10 hard, constructed from the prompt through a human-in-the-loop process. The released annotation procedure requires every question to be unambiguous, visually verifiable, and consistent with the prompt specification. We use the 20 questions as $\mathcal{C}$ and instantiate the two priority groups of the hard and easy subsets.

IGenBench. The question set for a prompt p is $\mathcal{Q}(p) = \mathcal{Q}_p(p) \cup \mathcal{Q}_e(p)$. The prompt-derived part $\mathcal{Q}_p$ is obtained by splitting the prompt into sentences and converting each sentence into a self-contained yes/no question, and accounts for 3,730 of the 5,259 released questions. The expert-informed part $\mathcal{Q}_e$ instantiates three general visualization requirements, namely data completeness, data ordering, and data encoding, and accounts for the remaining 1,529. We use $\mathcal{Q}(p)$ as $\mathcal{C}$ with a single priority group.

WISE. WISE provides no checklist, so we instantiate the prompt parser as $\mathcal{C} = \Phi(P)$, where Φ prompts Kimi-K3 to decompose the prompt into atomic, image-answerable requirements in a structured format. The prompt parser is instantiated with the following prompt.

Prompt Parser Prompt

```
KIMI_ENHANCE_SYSTEM = """You are a world knowledge expert
specializing in visual content understanding.

Your task: Given a short prompt that may contain implicit knowledge
(like cultural references, historical events, scientific concepts),
infer the intended subject and provide an enhanced prompt suitable
for image generation.

IMPORTANT: You must rely on your own knowledge. Do NOT reveal that
you're using search engines.
"""

KIMI_ENHANCE_USER_TEMPLATE = """Original prompt: "{original_prompt}"

Please provide (output as JSON):
{{
      "reasoning": "Your detailed thought process - what knowledge
is required to understand this prompt",
      "confidence": 0.0-1.0,  // Your confidence in the reasoning
      "enhanced_prompt": "Explicit, visual description for image
generation (can be a complete rewrite)",
      "verification_checklist": {{
            "primary_subject": "Main object/concept that MUST
appear",
            "key_features": ["feature1", "feature2", ...],
            "optional_context": "Optional scene/atmosphere"
         }}
   }}

Rules:
1. If confidence < 0.7, include "need_search" in your reasoning
2. Enhanced prompt should be explicit and visual (no implicit
knowledge)
3. Verification checklist is YOUR inference (not ground truth - you
don't have it)
4. You can completely rewrite the prompt if needed
"""
```

B BENCHMARK DETAILS

Table 6: Quantitative overview of the public benchmarks used in our evaluation. Final reporting follows each benchmark's official evaluator and metrics.

Benchmark	Scale	Coverage	Final eval. units	Metrics
BizGenEval	400 prompts	5 domains $\times$ 4 capabilities = 20 tasks	8,000 checklist judgments	Score, Accuracy
IGenBench	600 cases	30 infographic types in 6 categories	5,259 QA judgments	Q-ACC, I-ACC
WISE	1,000 prompts	3 domains, 25 subdomains, 6 report categories	3,000 dimension scores	WiScore

We evaluate STRIDE on three large-scale public benchmarks that provide complementary tests of infographic and structured visual generation. BizGenEval focuses on commercial visual documents with dense text, layout, attribute, and knowledge constraints. IGenBench targets text-to-infographic reliability, especially faithful data encoding and end-to-end constraint satisfaction. WISE evaluates knowledge-grounded generation from indirect semantic cues. Table 6 summarizes the scale, coverage, and evaluation protocol of each benchmark.

BizGenEval. BizGenEval (Li et al., 2026a) crosses five professional content domains (slides, charts, webpages, posters, and scientific figures) with four capability dimensions of text rendering, layout control, attribute binding, and knowledge-based reasoning. The prompt set contains 300 content-based prompts derived from real-world commercial visual materials and 100 knowledge-based prompts covering physics, chemistry, mathematics, history, and arts. Each prompt is paired with 20 task-specific verification questions, and final scores are computed from MLLM checklist judgments on easy and hard tracks. BizGenEval uses Gemini-3-Flash-Preview (Google DeepMind, 2025b) as its official evaluator.

BizGenEval's official code reports two closely related checklist-based metrics. Let q denote the number of questions in a track and let Q_{err} be the number of incorrectly answered questions. The (1) *checklist accuracy* is

$$\mathrm{Accuracy} = \frac{q - Q_{\mathrm{err}}}{q}. \tag{10}$$

The official leaderboard (2) *error score* applies a penalty to the number of errors,

$$\mathrm{Score} = \max\bigl(0, 1 - \alpha Q_{\mathrm{err}}\bigr), \tag{11}$$

where BizGenEval sets $\alpha = 0.2$. Thus, for a 10-question track, the score becomes zero once five or more questions are answered incorrectly. Following the leaderboard protocol, BizGenEval-related results report the official error score, unless otherwise stated.

IGenBench. IGenBench (Tang et al., 2026) contains 600 curated test cases spanning 30 infographic types, its evaluation decomposes each infographic specification into atomic yes/no verification questions. These questions cover 10 types of infographic elements (Title & Subtitle, Chart/Diagram Type, Decorative/Non-data Elements, Annotations & Callouts, Axes & Scales, Legend & Category Mapping, Data Marks, Data Completeness, Data Ordering, and Data Encoding). IGenBench uses Gemini-2.5-Pro (Comanici et al., 2025) as its official evaluator, reporting both Q-ACC, which measures individual constraint satisfaction, and I-ACC, which requires all questions for an infographic to be answered correctly. The official released dataset contains 5,259 atomic yes/no verification questions, including 3,730 prompt-derived questions and 1,529 expert-informed seed questions.

WISE. WISE (Niu et al., 2025) evaluates world-knowledge-informed image generation. Unlike benchmarks that use direct prompts, WISE avoids naming the target entity explicitly and instead describes it through clues, relations, or conditions. WISE contains 1,000 prompts across 25 subdomains in three major areas, including 400 cultural prompts, 300 spatio-temporal prompts, and 300 natural-science prompts, while the latter two are further divided into time, space, biology, physics, and chemistry categories. WISE uses WiScore as its metric and uses GPT-4o-2024-05-13 (Hurst et al., 2024) as its official evaluator. WiScore combines consistency, realism, and aesthetic quality, with default weights of 0.7, 0.2, and 0.1, respectively, and normalizes the weighted score to $[0, 1]$.

C ADVANTAGES OVER EXISTING AGENTIC METHODS

Table 7: Comparison of agentic methods on the mechanisms required for infographic refinement in the inference time. A, G and V for the agent backbone, base generator, and verifier, with O, P, and – denoting open-source or open-weight, proprietary, and not applicable.

Method	Venue	A / G / V	WebSearch	Rubric Feedback	Global Redraw	Local Edit	Iterative Refine.	Champion Admit
ReflectionFlow	25@ICCV	– / O / O	×	×	√	×	√	×
T2I-Copilot	25@ICCV	P / O / O	×	×	√	×	√	×
Ag3	25@ICML	P / P / P	×	×	√	×	√	×
Gen-Searcher	26@Arxiv	O / P / –	√	×	×	×	×	×
GenEvolve	26@Arxiv	O / P / –	√	×	×	×	×	×
Mind-Brush	26@SIGAsia	P / P / –	√	×	×	×	×	×
PromptEnhancer	26@CVPR	O / O / O	×	×	×	×	×	×
PaperBanana	26@ICML	P / P / P	√	×	√	×	√	×
AutoFigure	26@ICLR	P / P / P	×	×	√	√	√	×
STRIDE	**Ours**	**O / O / O**	√	√	√	√	√	√

Existing agentic methods can be partitioned into the following three categories:

(1) **Retrieval-Augmented Planning Methods.** A line of retrieval-augmented planning methods, exemplified by PromptEnhancer (Wang et al., 2025), Mind-Brush (He et al., 2026), Gen-Searcher (Feng et al., 2026), and GenEvolve (Chen et al., 2026), augments text-to-image generation by searching for implicit prompt content or retrieving reference images to condition the generator. These methods subsequently reorganize the prompt through a planning stage, yielding an enhanced prompt intended to improve generation quality. Nevertheless, these approaches are fundamentally limited to one-shot prompt rewriting. As a result, they not only fall short of enforcing the desired constraints, but can also compromise previously verified correct content.

(2) **Layout-Augmented Planning Methods.** Methods such as Ag3 (Hahn et al., 2024) and AutoFigure (Zhu et al., 2026b) augment generation with explicit layout planning, typically via editable graphs or SVGs, followed by fixed-round layout-guided refinement. However, their reliance on closed-source frontier models throughout generation and optimization obscures whether the gains stem from the agentic design or from the underlying models. Their extra layout-control components also limit compatibility with most open-source text-to-image generators.

(3) **Analysis-Evaluation Methods.** Methods represented by ReflectionFlow (Zhuo et al., 2025), T2I-Copilot (Chen et al., 2025), and PaperBanana (Zhu et al., 2026a) refine generated images under the guidance of a verifier that evaluates the current output. Yet the verifier feedback is typically not rubric-grounded, making it difficult to translate detected failures into precise and actionable corrections. In addition, their execution space is narrow, usually restricted to prompt revision or full regeneration. As a result, full-image regeneration can introduce regressions by altering already verified correct regions, leaving limited room for reliable improvement.

These existing systems improve over single-pass generation in many settings, but dense infographic generation requires additional mechanisms. In particular, a refinement loop must identify which rubrics failed, decide whether the failure should be addressed by global regeneration or localized editing, and prevent later revisions from degrading constraints that were already satisfied.

STRIDE addresses these limitations and demonstrates clear advantages. We compare these agentic methods in Table 7. For each representative method, we record whether it uses web search, rubric feedback as refinement state, global redrawing, local editing, iterative refinement, and champion admission, and we report the access profile of the agent backbone, base generator, and verifier.

For STRIDE, all components are open-weight, reducing attribution ambiguity and making the reported gains reflect the framework rather than proprietary model strength. The failed rubrics returned by dual-route evaluation guide retrieval, planning, and editing, allowing each refinement step to address the concrete errors present in the current image. During iterations, STRIDE operates in a scale-aware execution space, adaptively combining global regeneration with local editing so that the two serve complementary roles rather than conflicting alternatives. Finally, a $(1 + \lambda)$-evolution strategy accepts a proposal only when it strictly improves the current champion, ensuring monotonic and non-regressive refinement across rounds.

Table 8: Extended quantitative comparison on BizGenEval. Each entry reports the hard score followed by the easy score in parentheses. Rows are ordered by Venue and, within each year, by hard Average. Params denotes the parameters of the base generator. The proprietary-model results are reproduced unchanged from the official table of BizGenEval. Missing entries are denoted by "–".

Model	Venue	Params	Visual Domains					Capabilities				Average
			Slides	Webpage	Poster	Chart	Sci. Figure	Layout	Attribute	Text	Knowledge	
Proprietary models												
Nano-Banana-Pro	25@Google	–	82.2 (94.8)	77.5 (96.5)	76.5 (94.8)	73.0 (92.2)	74.2 (90.0)	72.2 (91.2)	65.6 (92.2)	86.4 (95.0)	82.6 (96.2)	76.7 (93.7)
Wan2.6-T2I	26@Alibaba	–	27.1 (56.5)	25.5 (67.0)	27.5 (62.5)	17.2 (48.8)	12.5 (58.8)	46.4 (80.6)	16.6 (60.6)	12.6 (52.6)	12.2 (41.0)	21.9 (58.7)
Seedream-5.0	26@ByteDance	–	54.5 (80.8)	47.0 (80.8)	50.7 (77.0)	46.0 (76.2)	45.8 (81.5)	67.6 (89.0)	42.4 (77.2)	43.4 (75.6)	41.8 (75.2)	48.8 (79.2)
Seedream-5.0-Pro	26@ByteDance	–	–	–	–	–	–	75.5 (87.6)	74.1 (87.3)	63.2 (81.1)	49.8 (75.3)	65.7 (82.8)
GPT-Image-1.5	26@OpenAI	–	40.8 (89.2)	41.0 (86.0)	42.0 (83.5)	28.2 (76.5)	27.8 (72.8)	51.6 (84.8)	25.8 (75.2)	40.4 (82.8)	26.0 (83.6)	35.9 (81.6)
Nano-Banana-2.0	26@Google	–	73.8 (95.8)	71.2 (94.5)	67.5 (91.2)	60.2 (89.2)	69.5 (92.0)	68.4 (91.0)	57.4 (91.6)	83.4 (94.6)	64.6 (93.0)	68.5 (92.5)
Open-weight models												
Qwen-Image	25@Alibaba	20B	3.5 (28.5)	2.5 (27.5)	5.8 (32.2)	2.2 (15.8)	0.0 (15.2)	10.4 (51.2)	0.2 (22.2)	0.6 (17.6)	0.0 (4.4)	2.8 (23.8)
FLUX.2-dev	25@BFL	32B	5.5 (43.2)	5.5 (48.8)	7.2 (48.2)	5.5 (33.5)	0.5 (36.5)	17.2 (67.8)	1.2 (49.2)	1.0 (43.0)	0.0 (8.2)	4.9 (42.0)
Qwen-Image-2512	25@Alibaba	20B	10.2 (45.0)	7.2 (48.0)	11.5 (47.0)	2.2 (28.0)	0.2 (37.0)	22.2 (70.6)	1.2 (47.8)	1.8 (39.2)	0.0 (6.4)	6.3 (41.0)
Z-Image	25@Z-Image	6B	12.2 (43.8)	6.2 (48.5)	12.0 (50.0)	8.8 (30.5)	1.8 (46.0)	26.8 (69.2)	2.6 (47.6)	2.8 (45.0)	0.6 (13.2)	8.2 (43.8)
HunyuanImage-3.0	25@Tencent	80B-A13B	19.3 (47.0)	21.0 (52.0)	19.8 (53.5)	2.0 (18.8)	2.8 (29.0)	27.8 (65.0)	13.8 (53.6)	10.2 (39.6)	0.0 (2.0)	13.0 (40.1)
Emu3.5	25@BAAI	34.1B	14.5 (44.8)	20.0 (48.8)	20.3 (53.0)	4.8 (20.3)	6.5 (34.0)	30.4 (63.4)	14.2 (52.6)	7.0 (33.6)	1.2 (11.0)	13.2 (40.2)
ERNIE-Image	26@Baidu	8B	32.8 (57.0)	41.5 (65.6)	35.6 (62.8)	10.5 (35.3)	19.7 (55.4)	51.8 (78.7)	31.8 (73.9)	28.3 (65.1)	0.1 (3.1)	28.0 (55.2)
Infographic-V2	26@SenseTime	8B	41.5 (64.2)	45.5 (66.8)	43.5 (63.7)	41.5 (59.2)	33.5 (64.8)	58.8 (91.4)	42.6 (73.8)	63.0 (84.4)	0.0 (5.4)	41.1 (63.7)
Agentic framework												
PE (ERNIE-Image)	26@Baidu	8B-3B	30.5 (58.2)	33.5 (59.5)	30.0 (50.7)	11.8 (40.0)	15.2 (46.2)	39.2 (56.6)	27.4 (61.8)	26.0 (60.4)	4.2 (25.0)	24.2 (50.9)
GenEvolve (ERNIE-Image)	26@Arxiv	8B	26.8 (48.5)	39.3 (63.8)	28.3 (49.0)	9.8 (39.0)	10.3 (41.3)	42.8 (65.6)	27.0 (58.2)	18.8 (54.6)	2.8 (14.8)	22.9 (48.3)
GenEvolve (ERNIE-Image-PE)	26@Arxiv	8B-3B	25.5 (52.5)	30.8 (52.8)	24.2 (50.0)	9.8 (36.8)	11.8 (46.8)	35.6 (59.6)	20.4 (59.2)	19.4 (50.6)	6.2 (21.6)	20.4 (47.8)
Gen-Searcher (ERNIE-Image)	26@Arxiv	8B	31.8 (63.5)	36.5 (63.0)	37.2 (65.0)	11.0 (40.8)	22.0 (60.5)	47.6 (78.0)	30.2 (67.6)	23.6 (62.6)	9.4 (26.0)	27.7 (58.6)
Gen-Searcher (Infographic-V2)	26@Arxiv	8B	46.5 (76.2)	41.0 (71.8)	45.0 (67.8)	41.8 (62.7)	41.8 (74.2)	58.2 (85.4)	38.6 (75.6)	58.2 (78.8)	17.8 (42.4)	43.2 (70.5)
PromptEnhancer-7B (ERNIE-Image)	26@CVPR	8B	15.5 (42.8)	17.2 (46.8)	14.8 (42.2)	9.2 (21.2)	3.2 (23.2)	32.4 (65.4)	9.0 (44.0)	6.2 (26.0)	0.4 (5.6)	12.0 (35.2)
PromptEnhancer-32B (ERNIE-Image)	26@CVPR	8B	18.5 (47.8)	17.5 (49.8)	18.0 (51.7)	12.0 (37.8)	5.5 (39.0)	35.4 (73.4)	12.6 (55.8)	8.0 (41.8)	1.2 (9.8)	14.3 (45.2)
PromptEnhancer-7B (Infographic-V2)	26@CVPR	8B	13.5 (41.8)	9.5 (34.0)	13.2 (37.5)	10.0 (24.2)	2.8 (27.3)	30.6 (64.8)	5.8 (38.2)	2.2 (18.6)	0.6 (10.2)	9.8 (33.0)
PromptEnhancer-32B (Infographic-V2)	26@CVPR	8B	17.2 (46.5)	14.2 (42.8)	12.8 (49.0)	19.0 (44.2)	6.8 (41.2)	36.0 (80.6)	10.0 (49.0)	8.6 (38.0)	1.4 (11.4)	14.0 (44.8)
STRIDE (ERNIE-Image)	**Ours**	8B	61.3 (79.2)	60.2 (83.2)	54.5 (74.5)	37.0 (62.3)	42.5 (69.2)	64.4 (80.8)	50.8 (82.6)	55.2 (79.8)	34.0 (51.6)	51.1 (73.7)
STRIDE (Infographic-V2)	**Ours**	8B	**73.2 (86.2)**	**69.5 (88.8)**	**74.5 (85.5)**	**67.5 (78.2)**	**63.0 (82.2)**	**79.2 (96.0)**	**65.8 (85.6)**	**79.2 (88.4)**	**54.0 (66.8)**	**69.5 (84.2)**

D EXTENDED EXPERIMENTS

D.1 EXTENDED MAIN RESULTS

Table 8 extends the BizGenEval results by reporting both hard and easy scores for each visual domain, capability, and their average. This provides a more comprehensive view of STRIDE's performance: the hard track reflects performance on more demanding requirements, while the easy track complements it by showing how well each method handles less challenging cases.

On the hard track, STRIDE consistently improves its base generator across all reported domains and capabilities. With ERNIE-Image, STRIDE raises the average score from 28.0 to 51.1, a gain of 23.1 points. With Infographic-V2, it raises the score from 41.1 to 69.5, a gain of 28.4 points. The improvements also extend to the easy track, where the corresponding averages increase from 55.2 to 73.7 (+18.5) and from 63.7 to 84.2 (+20.5), respectively. For example, with Infographic-V2, the hard scores for layout, attribute, text, and knowledge improve by 20.4, 23.2, 16.2, and 54.0 points, respectively. These results show that the gains are not confined to visual presentation, but also include capabilities related to satisfying specified attributes, rendering text, and incorporating relevant knowledge.

STRIDE also substantially outperforms the listed prior agentic frameworks when paired with the same base generator. On the hard track, its average exceeds the strongest corresponding baseline by 23.4 points with ERNIE-Image (51.1 vs. 27.7) and by 26.3 points with Infographic-V2 (69.5 vs. 43.2). The advantage is also present on the easy track, with gains of 15.1 points (73.7 vs. 58.6) and 13.7 points (84.2 vs. 70.5), respectively. Thus, the improvements are not limited to a single difficulty track or a single base generator.

Finally, STRIDE (Infographic-V2) surpasses several proprietary models on the hard average, including Seedream-5.0 (48.8), Seedream-5.0-Pro (65.7), and Nano-Banana-2.0 (68.5), with a score of 69.5. It also exceeds these models on the easy average where applicable: its score of 84.2 is higher than Seedream-5.0 (79.2) and Seedream-5.0-Pro (82.8), though lower than Nano-Banana-2.0 (92.5). These comparisons indicate that an open-weight generator combined with STRIDE can outperform a number of proprietary systems, particularly on the more demanding hard track.

Table 9: Extended STRIDE ablation study on BizGenEval and WISE benchmarks.

#	Components					BizGenEval									**Average**	WISE
	Lead	Comp.	Ret.	Perc.	Iters.	Slides	Webpage	Poster	Chart	Sci. Fig.	Layout	Attr.	Text	Know.	**Score**	Overall
(1)		Best-of-1			1 iter	32.8 (57.0)	41.5 (65.6)	35.6 (62.8)	10.5 (35.3)	19.7 (55.4)	51.8 (78.7)	31.8 (73.9)	28.3 (65.1)	0.1 (3.1)	28.0 (55.2)	0.58
(2)		Best-of-6			1 iter	39.0 (58.2)	48.5 (69.0)	40.2 (64.8)	19.0 (44.8)	26.8 (58.0)	60.6 (81.0)	41.4 (80.4)	36.4 (70.6)	0.4 (3.8)	34.7 (59.0)	–
(3)	✓	✓	–	–	1 iter	44.5 (65.0)	53.5 (71.2)	45.5 (64.5)	21.2 (46.5)	37.5 (69.0)	63.8 (81.6)	43.0 (80.4)	39.4 (69.2)	15.6 (21.8)	40.4 (63.2)	0.89
(4)	✓	✓	–	–	2 iters	44.5 (64.0)	54.2 (70.5)	44.5 (66.0)	22.2 (48.8)	38.7 (69.8)	64.4 (81.6)	43.2 (80.6)	40.8 (68.8)	15.0 (24.2)	40.8 (63.8)	–
(5)	✓	✓	–	–	3 iters	48.5 (66.5)	53.2 (69.5)	44.8 (65.0)	21.2 (49.0)	38.0 (68.8)	63.8 (80.6)	45.4 (78.4)	40.6 (70.0)	14.8 (26.0)	41.1 (63.7)	–
(6)	✓	–	✓	–	1 iter	52.5 (74.2)	59.2 (80.8)	51.0 (74.2)	34.0 (59.0)	37.8 (67.8)	63.4 (81.2)	40.8 (80.8)	53.8 (78.8)	29.6 (44.0)	46.9 (71.2)	–
(7)	✓	–	✓	–	2 iters	55.2 (76.8)	59.0 (82.2)	52.0 (76.0)	36.5 (61.3)	39.8 (69.5)	62.2 (82.2)	40.4 (81.4)	57.4 (80.2)	34.0 (48.8)	48.5 (73.2)	–
(8)	✓	–	✓	–	3 iters	57.5 (79.8)	60.5 (82.2)	52.0 (76.5)	38.8 (63.0)	42.5 (74.0)	64.4 (83.6)	41.4 (80.8)	60.4 (83.6)	34.8 (52.4)	50.2 (75.1)	–
(9)	✓	✓	✓	–	1 iter	52.8 (73.2)	55.5 (77.0)	51.5 (71.0)	30.8 (52.8)	37.8 (66.2)	63.2 (81.8)	45.6 (81.2)	49.4 (75.6)	24.4 (33.6)	45.6 (68.0)	0.89
(10)	✓	✓	✓	–	2 iters	55.5 (73.2)	57.2 (81.2)	51.7 (72.8)	31.0 (55.0)	39.0 (68.2)	63.4 (81.8)	46.6 (80.8)	50.4 (75.8)	27.2 (42.0)	46.9 (70.1)	–
(11)	✓	✓	✓	–	3 iters	57.0 (74.2)	57.0 (82.2)	54.5 (74.0)	34.8 (59.8)	41.5 (68.0)	64.2 (80.8)	49.4 (82.4)	51.0 (77.2)	31.2 (46.2)	48.9 (71.7)	–
(12)	✓	–	✓	✓	1 iter	52.8 (74.0)	59.2 (81.0)	52.0 (74.0)	33.5 (58.0)	37.8 (68.5)	64.2 (81.0)	40.8 (80.2)	53.4 (78.6)	29.8 (44.6)	47.0 (71.1)	–
(13)	✓	–	✓	✓	2 iters	54.8 (76.5)	59.8 (83.0)	54.2 (75.8)	35.0 (61.3)	38.8 (68.5)	61.6 (81.8)	43.0 (81.2)	55.4 (80.4)	34.0 (48.6)	48.5 (73.0)	–
(14)	✓	–	✓	✓	3 iters	58.2 (80.0)	61.8 (83.8)	54.2 (76.2)	36.5 (62.5)	41.5 (73.0)	63.6 (83.2)	44.0 (80.4)	59.0 (84.0)	35.2 (52.8)	50.4 (75.1)	–
(15)	✓	✓	✓	✓	1 iter	54.2 (74.0)	56.5 (77.2)	51.2 (71.0)	30.8 (52.8)	39.0 (67.2)	63.2 (81.8)	45.8 (81.2)	51.6 (76.4)	24.8 (34.4)	46.4 (68.5)	0.91
(16)	✓	✓	✓	✓	2 iters	56.0 (74.5)	59.0 (82.0)	51.5 (72.8)	34.0 (57.0)	40.2 (69.2)	63.6 (81.8)	47.8 (81.0)	53.8 (77.6)	27.4 (44.0)	48.1 (71.1)	0.92
(17)	✓	✓	✓	✓	3 iters	61.3 (79.2)	60.2 (83.2)	54.5 (74.5)	37.0 (62.3)	42.5 (69.2)	64.4 (80.8)	50.8 (82.6)	55.2 (79.8)	34.0 (51.6)	51.1 (73.7)	0.92

Table 10: Per-iteration scale-aware sub-agent preference of the lead editor, reported as composer / retoucher percentages on the BizGenEval with two generators. The number of rounds is given in parentheses.

Setup	**Dimension**	**Iter 1**	**Iter 2**	**Iter 3**	**Overall**
ERNIE-Image	Layout	97.2 / 2.8 (72)	92.7 / 7.3 (55)	89.6 / 10.4 (48)	93.7 / 6.3 (175)
	Attribute	82.7 / 17.3 (81)	77.3 / 22.7 (75)	81.9 / 18.1 (72)	80.7 / 19.3 (228)
	Text	9.2 / 90.8 (87)	17.6 / 82.4 (68)	23.7 / 76.3 (59)	15.9 / 84.1 (214)
	Knowledge	55.1 / 44.9 (98)	53.4 / 46.6 (88)	42.2 / 57.8 (83)	50.6 / 49.4 (269)
	Total	58.9 / 41.1 (338)	58.7 / 41.3 (286)	57.6 / 42.4 (262)	**58.5 / 41.5** (886)
Infographic-V2	Layout	94.4 / 5.6 (72)	93.4 / 6.6 (61)	96.2 / 3.8 (52)	94.6 / 5.4 (185)
	Attribute	82.7 / 17.3 (81)	82.7 / 17.3 (75)	85.9 / 14.1 (71)	83.7 / 16.3 (227)
	Text	11.5 / 88.5 (87)	15.5 / 84.5 (71)	31.6 / 68.4 (57)	18.1 / 81.9 (215)
	Knowledge	71.6 / 28.4 (95)	68.3 / 31.7 (82)	74.2 / 25.8 (66)	71.2 / 28.8 (243)
	Total	63.6 / 36.4 (335)	64.4 / 35.6 (289)	72.4 / 27.6 (246)	**66.3 / 33.7** (870)

D.2 EXTENDED ABLATION RESULTS

Table 9 provides a more detailed view of the component and iteration ablations, reporting both hard and easy scores across BizGenEval dimensions, together with WISE scores (WiScore) where available. These results complement the main-text ablation by showing how performance evolves under different component configurations and refinement budgets.

Benefits of Combining Components and Iterating. Combining the composer and retoucher yields higher hard average scores than using the composer alone at each matched iteration budget: 45.6 vs. 40.4 after one iteration (Table 9 (9):(3)), 46.9 vs. 40.8 after two (Table 9 (10):(4)), and 48.9 vs. 41.1 after three (Table 9 (11):(5)). Adding the perceptor to the lead-editor–retoucher configuration also improves the three-iteration hard average score from 50.2 to 50.4 (Table 9 (14):(8)), with gains in slides, webpage, and scientific figure scores. With all components enabled (Table 9 (15)-(16)-(17)), the hard average score increases from 46.4 after one iteration to 48.1 after two and 51.1 after three, and the corresponding Easy Average increases from 68.5 to 71.1 and 73.7. The WISE rows show transfer beyond checklist benchmarks, STRIDE improves the score from 0.58 under the Best-of-1 baseline to 0.92 after two iterations (Table 9 (16):(1)), further demonstrating the efficiency of the STRIDE framework.

Scale-Aware Sub-Agent Preferences. Table 10 further breaks down the lead editor's preferences for the composer and retoucher by generator, dimension, and iteration. The preferences vary substantially across dimensions, rather than consistently favoring a single sub-agent. For both generators, the composer is strongly preferred for layout (93.7% overall for ERNIE-Image and 94.6% for Infographic-V2) and is also preferred for attribute and knowledge. In contrast, the retoucher is preferred for text, accounting for 84.1% of the decisions with ERNIE-Image and 81.9% with

Table 11: Extended quantitative comparison on BizGenEval with oracle experiments. Each entry reports the hard score followed by the easy score in parentheses. Rows are ordered by Venue and, within each year, by hard average. Params denotes the parameters of the base generator. The proprietary-model results are reproduced unchanged from the official table of BizGenEval benchmark. Missing entries are denoted by "–".

Model	Venue	Params	Visual Domains					Capabilities				Average
			Slides	Webpage	Poster	Chart	Sci. Figure	Layout	Attribute	Text	Knowledge	
Proprietary models												
Nano-Banana-Pro	25@Google	–	82.2 (94.8)	77.5 (96.5)	76.5 (94.8)	73.0 (92.2)	**74.2 (90.0)**	72.2 (91.2)	65.6 (92.2)	86.4 (95.0)	**82.6 (96.2)**	76.7 (93.7)
Wan2.6-T2I	26@Alibaba	–	27.1 (56.5)	25.5 (67.0)	27.5 (62.5)	17.2 (48.8)	12.5 (58.8)	46.4 (80.6)	16.6 (60.6)	12.6 (52.6)	12.2 (41.0)	21.9 (58.7)
Seedream-5.0	26@ByteDance	–	54.5 (80.8)	47.0 (80.8)	50.7 (77.0)	46.0 (76.2)	45.8 (81.5)	67.6 (89.0)	42.4 (77.2)	43.4 (75.6)	41.8 (75.2)	48.8 (79.2)
Seedream-5.0-Pro	26@ByteDance	–	–	–	–	–	–	75.5 (87.6)	**74.1 (87.3)**	63.2 (81.1)	49.8 (75.3)	65.7 (82.8)
GPT-Image-1.5	26@OpenAI	–	40.8 (89.2)	41.0 (86.0)	42.0 (83.5)	28.2 (76.5)	27.8 (72.8)	51.6 (84.8)	25.8 (75.2)	40.4 (82.8)	26.0 (83.6)	35.9 (81.6)
Nano-Banana-2.0	26@Google	–	73.8 (95.8)	71.2 (94.5)	67.5 (91.2)	60.2 (89.2)	69.5 (92.0)	68.4 (91.0)	57.4 (91.6)	83.4 (94.6)	64.6 (93.0)	68.5 (92.5)
Open-weight models												
ERNIE-Image	26@Baidu	8B	32.8 (57.0)	41.5 (65.6)	35.6 (62.8)	10.5 (35.3)	19.7 (55.4)	51.8 (78.7)	31.8 (73.9)	28.3 (65.1)	0.1 (3.1)	28.0 (55.2)
Infographic-V2	26@SenseTime	8B	41.5 (64.2)	45.5 (66.8)	43.5 (63.7)	41.5 (59.2)	33.5 (64.8)	58.8 (91.4)	42.6 (73.8)	63.0 (84.4)	0.0 (5.4)	41.1 (63.7)
Agentic framework												
STRIDE (ERNIE-Image)	**Ours**	8B	61.3 (79.2)	60.2 (83.2)	54.5 (74.5)	37.0 (62.3)	42.5 (69.2)	64.4 (80.8)	50.8 (82.6)	55.2 (79.8)	34.0 (51.6)	51.1 (73.7)
STRIDE (Infographic-V2)	**Ours**	8B	73.2 (86.2)	69.5 (88.8)	74.5 (85.5)	67.5 (78.2)	63.0 (82.2)	79.2 (96.0)	65.8 (85.6)	79.2 (88.4)	54.0 (66.8)	69.5 (84.2)
Oracle												
STRIDE (ERNIE-Image)	**Ours**	8B	70.2 (87.8)	76.0 (93.2)	70.5 (86.8)	47.8 (68.0)	57.5 (82.5)	71.8 (87.4)	63.6 (86.2)	67.2 (88.2)	54.8 (72.8)	64.4 (83.7)
STRIDE (Infographic-V2)	**Ours**	8B	**81.2 (91.5)**	**80.8 (95.2)**	**78.8 (89.2)**	**75.0 (87.2)**	70.5 (90.2)	**85.2 (97.2)**	73.2 (90.4)	**88.4 (94.2)**	62.2 (81.0)	**77.2 (90.7)**

Table 12: Extended quantitative comparison on IGenBench with oracle experiments. Params denotes the parameters of the base generator. The proprietary-model results are reproduced unchanged from the official IGenBench.

Model	Venue	Params	Question Type Q-ACC										Overall	
			Comp.	Enc.	Order	Marks	Anno.	Axes	Leg.	Chart	Title	Deco.	Q-ACC	I-ACC
Proprietary models														
Image-01	25@MiniMax	–	0.01	0.05	0.04	0.10	0.10	0.14	0.03	0.22	0.14	0.47	0.13	0.00
P-Image	25@PrunaAI	–	0.08	0.15	0.19	0.27	0.36	0.28	0.54	0.43	0.58	0.68	0.34	0.00
Nano-Banana	25@Google	–	0.18	0.31	0.27	0.44	0.54	0.57	0.52	0.60	0.65	0.81	0.48	0.02
Nano-Banana-Pro	25@Google	–	0.84	**0.86**	0.90	**0.87**	0.93	0.93	**0.96**	0.92	0.98	0.94	**0.90**	**0.49**
GPT-Image-1.5	26@OpenAI	–	0.38	0.48	0.44	0.57	0.50	0.54	0.57	0.68	0.60	0.80	0.55	0.12
Seedream-4.5	26@ByteDance	–	0.34	0.37	0.47	0.48	0.70	0.70	0.81	0.68	0.95	0.84	0.61	0.06
Seedream-5.0-Pro	26@ByteDance	–	0.64	0.47	0.71	0.73	0.77	0.85	0.83	0.83	0.86	0.88	0.73	0.17
GPT-Image-2.0	26@OpenAI	–	**0.89**	0.61	**0.93**	**0.87**	**0.94**	**0.98**	0.95	**0.93**	**0.99**	**0.95**	0.88	0.38
Qwen-Image-3.0	26@Alibaba	–	0.65	0.43	0.77	0.77	0.84	0.84	0.87	0.85	0.80	0.88	0.74	0.13
Open-weight models														
ERNIE-Image	26@Baidu	8B	0.17	0.16	0.25	0.34	0.43	0.47	0.61	0.55	0.57	0.81	0.41	0.01
Infographic-V2	26@SenseTime	8B	0.54	0.49	0.69	0.66	0.79	0.80	0.85	0.80	0.94	0.88	0.72	0.17
Agentic framework														
STRIDE (ERNIE-Image)	**Ours**	8B	0.30	0.32	0.41	0.52	0.60	0.63	0.73	0.70	0.67	0.89	0.55	0.03
STRIDE (Infographic-V2)	**Ours**	8B	0.70	0.72	0.81	0.78	0.90	0.89	0.92	0.91	0.98	0.94	0.84	0.38
Oracle														
STRIDE (Infographic-V2)	**Ours**	8B	0.71	0.74	0.82	0.80	0.92	0.90	0.92	0.92	**0.99**	0.94	0.85	0.40

Infographic-V2. This pattern is consistent with the different strengths of the two routes: broader composition changes are more often selected for layout-level issues, while localized editing is more often selected for text-related corrections.

The preference distributions also vary across iterations and generators. For ERNIE-Image, the overall composer preference remains relatively stable across the three rounds (58.9%, 58.7%, and 57.6%), while for Infographic-V2 it rises from 63.6% in the first round to 72.4% in the third. The dimension-level trends are not uniform: for example, the retoucher remains the majority choice for text in every round for both generators, whereas the composer remains strongly preferred for layout. These differences indicate that the lead editor's routing behavior depends on the generator, the error dimension, and the refinement round, rather than following a fixed composer-only or retoucher-only policy. Together with the improved results of the combined system, these preference patterns provide evidence for the scale-aware operation of STRIDE.

D.3 EXTENDED ORACLE EXPERIMENT RESULTS

To assess the model adaptability and further improvement potential of STRIDE, we replace only its perceptor and verifier with the proprietary models used by each benchmark, while keeping the open-

Table 13: Quantitative comparison on WISE. We report WiScore across six knowledge domains and the overall score. Best and second-best results are shown in bold and underline, respectively. The proprietary-model results are reproduced unchanged from the official WISE.

Model	Venue	Params	Cult.	Time	Space	Bio.	Phys.	Chem.	Overall
Proprietary models									
Seedream 4.0	25@ByteDance	–	0.78	0.73	0.85	0.79	0.84	0.67	0.78
GPT-4o / GPT-Image-1	25@OpenAI	–	0.81	0.71	0.89	0.83	0.79	0.74	0.80
Nano-Banana-Pro	25@Google	–	0.89	0.80	0.89	0.88	0.86	0.85	0.87
Open-weight models									
FLUX.1-dev	24@BFL	12B	0.48	0.58	0.62	0.42	0.51	0.35	0.50
Hunyuan-Image 3.0	25@Tencent	80B-A13B	0.58	0.57	0.70	0.56	0.63	0.31	0.57
Qwen-Image	25@Alibaba	20B	0.62	0.63	0.77	0.57	0.75	0.40	0.62
LongCat-Image	25@Meituan	6B	0.66	0.61	0.72	0.66	0.72	0.49	0.65
BAGEL+CoT	25@ByteDance	7B	0.76	0.69	0.75	0.65	0.75	0.58	0.70
NextFlow-RL	26@NextFlow	–	0.63	0.63	0.77	0.58	0.67	0.39	0.62
SenseNova-U1-SFT	26@SenseTime	8B	0.65	0.66	0.82	0.68	0.81	0.66	0.69
NEO-unify-CoT	26@SenseTime	8B	0.73	0.67	0.79	0.70	0.75	0.66	0.72
DeepGen1.0	26@DeepGen	–	0.72	0.81	0.70	0.67	0.82	0.66	0.73
ERNIE-Image	26@Baidu	8B	0.54	0.59	0.78	0.55	0.66	0.35	0.58
SenseNova-U1-SFT	26@SenseTime	8B-A3B	0.73	0.69	0.80	0.73	0.82	0.69	0.74
SenseNova-U1-SFT-CoT	26@SenseTime	8B	0.78	0.73	0.82	0.80	0.85	0.85	0.78
SenseNova-U1-SFT-CoT	26@SenseTime	8B-A3B	0.81	0.77	0.84	0.81	0.84	0.82	0.81
Agentic framework									
GenAgent (FLUX.1-dev)	26@arXiv	12B	0.78	0.67	0.78	0.72	0.77	0.55	0.72
Gen-Searcher (Qwen-Image)	26@arXiv	20B	0.80	0.71	0.82	0.76	0.74	0.75	0.77
Mind-Brush (Qwen-Image-Edit-2511)	26@SIGGRAPH Asia	20B	0.83	0.69	0.84	0.71	0.85	0.68	0.78
GenEvolve (Qwen-Image-Edit)	26@arXiv	–	0.84	0.74	0.87	0.83	0.81	0.83	0.82
PE (ERNIE-Image)	26@Baidu	–	0.61	0.65	0.81	0.59	0.74	0.59	0.65
PromptEnhancer-32B (ERNIE-Image)	26@CVPR	8B	0.81	0.77	0.87	0.86	0.85	0.74	0.81
STRIDE (Qwen-Image-2512)	Ours	20B	**0.93**	0.84	**0.93**	0.89	0.94	0.90	0.91
STRIDE (ERNIE-Image)	Ours	8B	0.92	**0.87**	**0.93**	**0.92**	**0.95**	**0.92**	**0.92**

weight generators and other framework components unchanged. Specifically, in oracle experiments, STRIDE uses Gemini-3-Flash-Preview (Google DeepMind, 2025b) as the perceptor and verifier on BizGenEval and Gemini-2.5-Pro (Comanici et al., 2025) for these roles on IGenBench, while all other components remain open-weight.

As shown in Tables 11 and 12, this stronger model configuration further improves both generators on BizGenEval: the hard average score increases from 51.1 to 64.4 with ERNIE-Image and from 69.5 to 77.2 with Infographic-V2. On IGenBench, using Infographic-V2, it also raises Q-ACC/I-ACC from 0.84/0.38 to 0.85/0.40. These results show that STRIDE's refinement process can benefit from different perception and verification backbones across benchmarks and generators, revealing additional headroom for the same open-weight generators without changing the framework itself. We view this headroom as evidence of STRIDE's extensibility: improvements in the models guiding refinement can translate into further gains from existing open-weight generators.

D.4 Extended Model Scalability Results on WISE

We further evaluate STRIDE on WISE to test whether the proposed refinement framework scales beyond infographic benchmarks with released checklists. Unlike BizGenEval and IGenBench, WISE does not provide explicit verification rubrics. Therefore, STRIDE constructs image-answerable constraints from the prompt using the parser Φ (Appendix A) and applies the same verifier-guided refinement procedure.

Table 13 shows that STRIDE achieves the best overall WiScore. STRIDE (ERNIE-Image) obtains 0.92 and STRIDE (Qwen-Image-2512) obtains 0.91, outperforming the strongest prior agentic baseline, GenEvolve (Chen et al., 2026) (0.82), by 0.10 and 0.09, respectively. They also exceed the strongest proprietary model, Nano-Banana-Pro (Google DeepMind, 2025c) (0.87), despite using open-weight generators with an all-open-weight agent stack. The gains are consistent across domains: STRIDE (ERNIE-Image) scores at least 0.87 in every category and achieves the best result in Time, Biology, Physics, and Chemistry, improving ERNIE-Image's standalone score from 0.58 to 0.92. And STRIDE (Qwen-Image-2512) leads in Culture and Space.

Table 14: Sampling protocol for the human–MLLM evaluation study. For each prompt, we retain paired outputs from the frozen generator and STRIDE under the same generator and prompt.

Benchmark	Generator	Prompt pairs	Images	Human rubric
BizGenEval	ERNIE-Image	25	50	25×10 Hard questions
BizGenEval	Infographic-V2	25	50	25×10 Hard questions
IGenBench	ERNIE-Image	25	50	All released atomic questions
IGenBench	Infographic-V2	25	50	All released atomic questions
Total	–	**100**	**200**	–

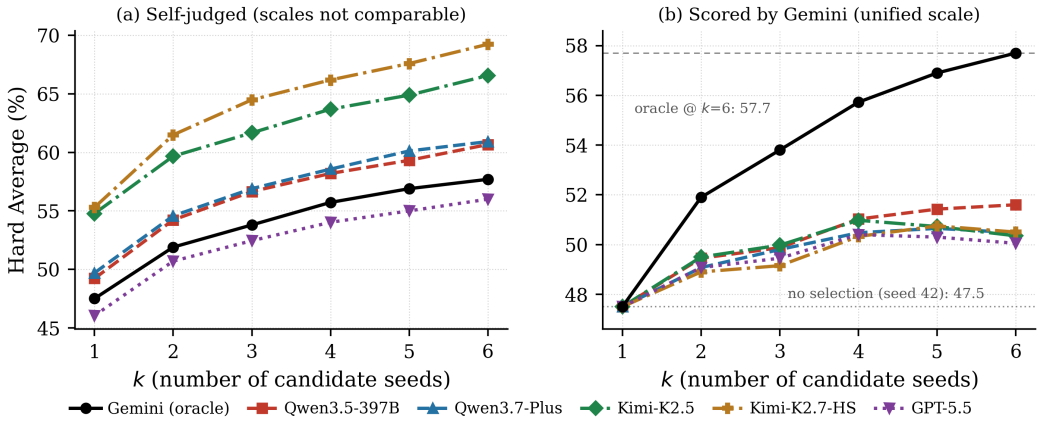

Figure 3: Best-of-K selection on BizGenEval over 400 items on the hard track. (a) Each verifier scores its own pick. (b) The same picks rescored by Gemini-3-Flash-Preview. Hard average values are mean per-item checklist accuracy on the hard track, whcih differ from error score (Appendix B).

These results suggest that STRIDE's benefit is not limited to checklist-based infographic evaluation. By dynamically instantiating rubrics from prompts, the same scale-aware refinement loop also improves knowledge-grounded generation where the target entity is only indirectly specified. Figure 7 compares STRIDE with contemporary agentic methods. STRIDE produces more consistent, realistic, and aesthetically appealing results, further illustrating the framework's scalability.

E HUMAN-MLLM STUDIES SETUP

Study Objectives and Sampling. This study assesses two aspects of our evaluation: agreement between automated evaluators and human judgments, and whether the improvement of STRIDE over its frozen generator is preserved under human evaluation. As summarized in Table 14, we sample 25 prompts for each combination of benchmark (BizGenEval or IGenBench) and generator (ERNIE-Image or Infographic-V2). Each prompt contributes two images: the frozen-generator output and the final STRIDE output. The study therefore contains 100 prompt–generator pairs and 200 images. Pairing controls for prompt identity within each generator.

Evaluation Sources and Rubrics. Each image is evaluated using human judgments, the default Qwen3.5-397B-A17B verifier, and the corresponding official benchmark evaluator: Gemini-3-Flash-Preview for BizGenEval and Gemini-2.5-Pro for IGenBench. All three evaluation sources assess the same atomic questions for each image, and take the average of multiple evaluations. For BizGenEval, the human study uses the 10 released hard-track questions. For IGenBench, it uses all released atomic questions associated with the sampled prompt. Human evaluation is thus rubric-based rather than a holistic aesthetic preference assessment.

Table 15: Best-of-6 selection on BizGenEval over 400 items, reported as mean checklist accuracy on the hard track. Self is the verifier's score of its own pick. Gemini is that pick rescored by Gemini. Capture is the fraction of the Gemini gain from 47.5 to 57.7 that each verifier realizes. Leniency is Self minus Gemini at $k = 1$, where the images coincide. Inflation is Self minus Gemini at $k = 6$.

Selector	Self	Gemini	Capture	Leniency	Inflation
Gemini (Oracle)	57.7	57.7	100%	+0.0	+0.0
Qwen3.5-397B-A17B	60.7	51.6	40.2%	+1.7	+9.1
Qwen3.7-Plus	60.9	50.5	29.2%	+2.2	+10.4
Kimi-K2.7-HS	69.3	50.5	29.4%	+7.8	+18.8
Kimi-K2.5	66.6	50.4	27.9%	+7.3	+16.2
GPT-5.5	56.0	50.1	25.0%	−1.4	+5.9

F Diagnostic Experiments

F.1 Choice of Rubric-Based Verifier

To identify a suitable rubric-based verifier, we compare verifier candidates via best-of-k selection on all 400 BizGenEval items, using nested pools of $k \in \{1, \ldots, 6\}$ ERNIE-Image samples, totaling 2,400 images. Each verifier ranks the k images by its assigned mean checklist accuracy (Appendix B) and selects the top-ranked image. Figure 3(a) reports self-judged performance, where each verifier both selects and scores its own best-of-k image, while Figure 3(b) reports Gemini-scored performance, where Gemini-3-Flash-Preview re-evaluates the selected images across verifiers. For each non-Gemini judge, the gap between its self-score and Gemini rescore increases with k.

As shown in Table 15, Qwen3.5-397B-A17B achieves a Gemini-scored average checklist accuracy of 51.6 at $k = 6$ and captures 40.2% of Gemini's selection gain, outperforming even its proprietary counterpart, Qwen3.7-Plus (Qwen Team, 2026c). Other open-weight judges and GPT-5.5 (OpenAI, 2026a) capture only 25.0% to 29.4%, whereas Qwen3.5-397B-A17B maintains a capture rate near 40% from $k = 2$ to $k = 6$. To avoid rubric-agnostic or overly permissive judging, as observed for Kimi-K2.5 (Kimi et al., 2026b) and Kimi-K2.7-HS (Moonshot AI, 2026), and based on comparisons between open-weight and proprietary models, we adopt the open-weight Qwen3.5-397B-A17B as our default rubric-based verifier.

F.2 Choice of Perceptor

The perceptor is an important but replaceable component in STRIDE. Its role is not to decide whether a candidate should be admitted, as champion selection is still governed by the rubric-based verifier score. Instead, the perceptor provides open-form visual grounding for the lead editor and the retoucher, including visible text blocks, their bounding boxes, and whether the rendered text appears corrupted. This distinction matters because the retoucher can only repair a failure that has been localized. The choice of perception backbone therefore bounds the practical effectiveness of local editing, but it is not a fixed assumption of the framework.

Protocol. We select the default perceptor backbone through a small, stratified agreement study and practical closed-loop verification. The study uses 20 generated infographics sampled across the five BizGenEval domains and two base generators, ERNIE-Image and Infographic-V2. Each candidate perceptor backend is asked to return visible text blocks and normalized bounding boxes. Since generated images do not come with ground-truth text-box annotations, we use Gemini-3-Flash-Preview as a proxy reference after sanity-checking its localization behavior against a dedicated PP-OCRv5 server (Cui et al., 2026). PP-OCRv5 server is therefore included in the table as a cross-check/reference OCR baseline, rather than as the perception backbone selected for STRIDE. We report *agreement* with the reference, not absolute accuracy.

We use a box-first protocol. Candidate boxes are first matched to reference boxes by IoU, and the text metrics are computed only on matched pairs. Unmatched reference boxes reduce agreement recall (A-R), while unmatched candidate boxes reduce agreement precision (A-P). A-F1 is their harmonic mean. Mean IoU measures the spatial overlap of matched boxes. NED-sim is one minus

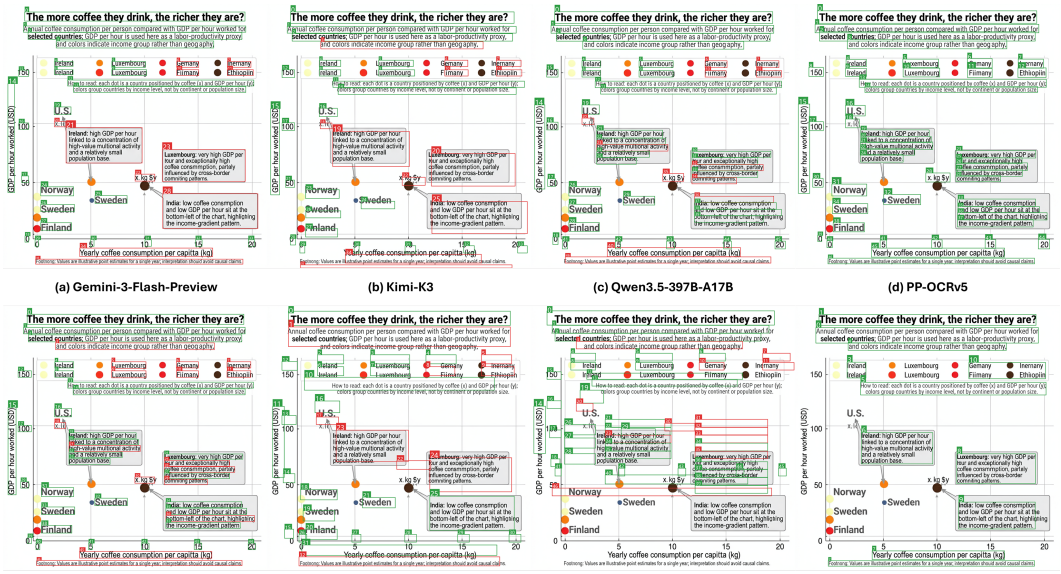

Figure 4: Qualitative comparison of perceptor outputs on generated infographics. Different perception backbones identify visible text blocks, localize them with bounding boxes, and assess whether the rendered text is garbled. Green boxes indicate regions that the perceptor identifies as correct, whereas red boxes indicate regions it identifies as incorrect. The comparison highlights the trade-off between geometric localization and contextual text understanding that motivates the use of an MLLM-based perceptor in STRIDE.

Table 16: Perception-backbone agreement with Gemini-3-Flash-Preview on 20 stratified generated infographics under a box-first protocol. PP-OCRv5 is reported as a cross-check/reference OCR baseline. Metrics measure agreement with the proxy reference, not absolute accuracy. Higher is better for A-R, A-P, A-F1, mean IoU, NED-sim, and exact; lower is better for anchor offset and latency.

Model	A-R	A-P	A-F1	mean IoU	NED-sim	exact	Anchor med. / P90	Med. latency
PP-OCRv5	**0.881**	**0.703**	**0.764**	0.729	0.899	0.590	**2.2 / 6.6**	44s
Kimi-K3	0.738	0.699	0.713	0.730	0.949	0.721	4.6 / 16.3	49s
Qwen3.5-397B-A17B	0.511	0.486	0.495	0.749	**0.957**	**0.751**	6.0 / 46.3	71s
Qwen3.8-Max	0.388	0.360	0.370	0.670	0.924	0.728	13.4 / 148.4	259s
LocateAnything-3B	0.345	0.535	0.322	**0.777**	–	–	–	**6s**
Kimi-K2.7-Code-HS	0.318	0.292	0.300	0.654	0.896	0.698	16.0 / 98.6	29s
Kimi-K2.7-Code	0.283	0.273	0.273	0.655	0.869	0.688	19.1 / 92.0	234s

normalized edit distance between the matched transcriptions, and exact is the fraction of matched pairs with identical text. We also report an anchor-offset diagnostic: for text strings that appear uniquely in both outputs, we measure the distance between the two box centers in the normalized 0–1000 coordinate system. Its median measures typical localization quality, while its P90 exposes long-tail failures. We provide a qualitative comparison of different perceptors in Figure 4.

Agreement Results. Table 16 shows that no single backend dominates every criterion. PP-OCRv5 has the strongest box coverage and the smallest anchor offsets, which confirms that the cross-check OCR baseline is a strong text localizer. However, its transcription agreement is substantially weaker than the strongest MLLM backends: its exact-match rate is 0.590, compared with 0.721 for Kimi-K3 (Kimi et al., 2026a) and 0.751 for Qwen3.5-397B-A17B (Qwen Team, 2026b). Qwen3.5-397B-A17B achieves the best matched-text similarity, but its recall and precision are much lower and its P90 anchor offset is nearly three times that of Kimi-K3, indicating a heavier tail of localization failures. Kimi-K3 provides the most balanced open-weight tradeoff: its A-F1 is close to the OCR cross-check baseline PP-OCRv5, its mean IoU is comparable, and its transcription agreement is

Table 17: Comparison of champion-evolution strategies on BizGenEval. Each entry reports the official error score as hard (easy). All non-oracle variants use the same refinement framework and differ only in how new champion candidates are selected and admitted. The $(1, 1)$ and $(1, 4)$ variants denote forced replacement, whereas the $(1 + 1)$ and $(1 + 4)$ variants retain the incumbent champion unless a better candidate is found. Oracle admission keeps Qwen3.5-397B-A17B within-round candidate selection unchanged but uses Gemini-3-Flash-Preview for admission; Full oracle uses Gemini-3-Flash-Preview for both candidate selection and admission.

Strategy	Content Domain					Capability				Average
	Slides	Webpage	Poster	Chart	Sci. Fig.	Layout	Attr.	Text	Know.	
Non-oracle strategies										
Random selection	45.0 (74.5)	59.8 (77.2)	47.5 (73.8)	21.3 (50.0)	28.5 (67.0)	54.6 (78.2)	40.0 (77.6)	46.4 (77.0)	20.6 (41.2)	40.4 (68.5)
$(1, 1)$ forced replacement	47.5 (71.8)	55.0 (77.2)	44.2 (65.0)	22.0 (43.5)	28.0 (60.8)	56.0 (75.2)	37.8 (73.8)	42.2 (72.8)	21.4 (32.8)	39.4 (63.7)
$(1 + 1)$-evolution strategy	53.0 (75.0)	60.2 (81.2)	50.3 (72.5)	29.8 (53.5)	34.5 (65.5)	62.6 (80.0)	43.6 (81.0)	49.2 (75.8)	26.8 (41.4)	45.6 (69.5)
$(1, 4)$ forced replacement	56.0 (76.2)	**62.0** (81.5)	49.3 (72.2)	27.8 (49.8)	32.0 (67.2)	59.2 (77.2)	42.4 (77.8)	50.4 (**79.8**)	29.6 (42.8)	45.4 (69.4)
$(1 + 4)$**-evolution strategy (ours)**	**61.3 (79.2)**	60.2 (**83.2**)	**54.5 (74.5)**	**37.0 (62.3)**	**42.5 (69.2)**	**64.4 (80.8)**	**50.8 (82.6)**	**55.2 (79.8)**	**34.0 (51.6)**	**51.1 (73.7)**
Oracle diagnostics										
Oracle admission (Qwen + Gemini)	63.0 (79.8)	70.5 (85.0)	60.5 (74.0)	42.3 (60.8)	48.3 (74.0)	68.2 (80.8)	56.8 (83.0)	61.0 (81.4)	41.6 (53.6)	56.9 (74.7)
Full oracle (Gemini + Gemini)	70.2 (87.8)	76.0 (93.2)	70.5 (86.8)	47.8 (68.0)	57.5 (82.5)	71.8 (87.4)	63.6 (86.2)	67.2 (88.2)	54.8 (72.8)	64.4 (83.7)

much higher than the conventional OCR baseline, including PP-OCRv5 and LocateAnything (Wang et al., 2026).

Why OCR Alone is Insufficient. The perceptor in STRIDE is not a generic OCR module. It must supply spatial evidence that is useful for deciding whether a failed rubric can be repaired locally. Conventional OCR and detection models (Zhou, 2024; Cui et al., 2026; Wang et al., 2026) are optimized to find text-like regions and transcribe them, but generated infographics often contain pseudo-text artifacts: glyphs that look like text locally but are misspelled, structurally broken, or semantically nonsensical in context. In such cases, as shown in Figure 4, a pure OCR system (like PP-OCRv5 in Figure 4 (d) or LocateAnything in Figure 4 (h)) can return precise boxes around corrupted glyphs and treat them as normal readable text. These false-positive text blocks are harmful for the retoucher, because they create repair targets that appear well localized while failing to represent reliable semantic evidence. An MLLM perceptor is better aligned with this use case because it jointly considers the box, the transcription, and the surrounding linguistic and visual context, including whether the text should be marked as garbled.

Backbone Choice. We choose Kimi-K3 as the default perceptor for reasons beyond single-call localization agreement. First, it offers the best overall balance between localization coverage and text understanding among the open-weight candidates in Table 16. Second, the rubric-based verifier in our default stack uses a Qwen-series model. Reusing the same model family for perception would increase the risk of correlated errors between the grounding signal and the verification signal, weakening the benefit of the dual-route design. Third, STRIDE does not rely on a single perception sample. Each refinement round aggregates 4 to 10 OCR/perception samples, and rollback triggers resampling in later rounds. This repeated-sampling design directly mitigates the long-tail localization failures observed in single-call diagnostics.

F.3 CHOICE OF EVOLUTION STRATEGY

Evolution Strategy Experiment Setup. Table 17 compares alternative champion-evolution strategies on BizGenEval. Each entry reports the official error score as hard (easy), with higher values indicating better performance. We use the Hard track as the primary evaluation protocol and report Easy-track scores for reference. The comparison separates two decisions in champion evolution. First, the *pick* decision selects the best candidate within the current new champion candidate set. Second, the *admission* decision determines whether the picked candidate should replace the incumbent champion. In the default $(1 + 4)$-evolution strategy, both decisions are made by the Qwen3.5-397B-A17B verifier. The oracle-admission variant keeps Qwen3.5-397B-A17B for within-round candidate selection but uses the Gemini-3-Flash-Preview evaluator for admission. The full-oracle variant uses Gemini-3-Flash-Preview for both candidate selection and admission.

Champion Candidates. Among the non-oracle variants, our default $(1 + 4)$-evolution strategy achieves the highest average hard score, reaching 51.1. Increasing the candidate count improves both forced-replacement and evolution strategies: the forced-replacement score increases from 39.4

for $(1, 1)$ to 45.4 for $(1, 4)$, while the evolution strategy score increases from 45.6 for $(1+1)$ to 51.1 for $(1 + 4)$. Thus, generating multiple proposals provides a substantial search benefit under both admission rules.

Incumbent Retention. Incumbent retention provides an additional and independent benefit. With one candidate, the evolution strategy improves the hard average from 39.4 for forced replacement $(1, 1)$ to 45.6 for $(1+1)$, a gain of 6.2 points. With four candidates, the corresponding improvement is 5.7 points, from 45.4 for forced replacement $(1, 4)$ to 51.1 for $(1+4)$. Therefore, the performance of the proposed strategy is not explained by increased candidate sampling alone: it also depends on rejecting non-improving updates and preserving a strong incumbent.

Selection Methods. Random selection reaches an average hard score of 40.4, which is substantially below the proposed $(1 + 4)$ strategy. It slightly outperforms forced replacement $(1, 1)$, which reaches 39.4, indicating that random replacement can occasionally benefit from stochastic diversity. However, it remains well below the strategies that combine multiple proposals with score-guided selection and incumbent retention. This result suggests that repeated stochastic modification alone is insufficient for reliable refinement.

Oracle Experiments in Evolution Strategy. We also conducted oracle evaluations in our evolution-strategy comparison. In Table 17, replacing only the Qwen-based admission decision with the Gemini-based benchmark evaluator increases the Hard average from 51.1 to 56.9, a gain of 5.8 points, while the corresponding Easy score increases from 73.7 to 74.7. Because within-round candidate selection remains Qwen-based in this setting, this improvement isolates the effect of the admission decision. The result indicates that the default verifier sometimes accepts inferior updates or rejects beneficial ones, with the largest Hard-score gains appearing in Knowledge (+7.6), Attribute (+6.0), and Text (+5.8).

The full-oracle variant, which uses Gemini for both within-round candidate selection and admission, further increases the Hard average to 64.4. This is 7.5 points above oracle admission and 13.3 points above the default $(1 + 4)$ strategy. Since admission is already controlled by Gemini in both oracle variants, the additional gap isolates errors in Qwen's within-round candidate ranking. The largest additional gain is observed for Knowledge, which improves from 41.6 under oracle admission to 54.8 under the full oracle. These results show that both candidate ranking and incumbent admission contribute to the remaining performance gap.

Overall, the comparison supports the $(1 + \lambda)$-evolution strategy as a central component of STRIDE. Multiple candidates improve the search space explored in each refinement round, while incumbent retention prevents the system from replacing a strong champion with a lower-quality proposal. The oracle results further show that the effectiveness of this mechanism is bounded by the quality of the verifier used for candidate ranking and admission.

G MORE QUALITATIVE RESULTS

Figures 5 and 6 present additional qualitative results on infographic generation. Across the shown examples, STRIDE produces more complete and coherent infographics than existing agentic methods, with better preservation of the requested layout, attributes, text, and semantic content. In contrast, the compared methods more frequently exhibit missing or distorted text, incorrect element arrangements, and unresolved prompt constraints. These examples further illustrate the benefit of STRIDE's rubric-guided diagnosis, scale-aware execution, and iterative champion refinement.

Figure 7 further evaluates the transferability of STRIDE on the general WISE task. Although WISE is not restricted to infographic-specific requirements, STRIDE generates results that more faithfully preserve the prompt-specified objects, relationships, and visual structure than the compared agentic methods. The consistent improvements across these examples suggest that STRIDE's refinement mechanism is not tied to a particular visual domain and can generalize to tasks with prompt-derived requirements.

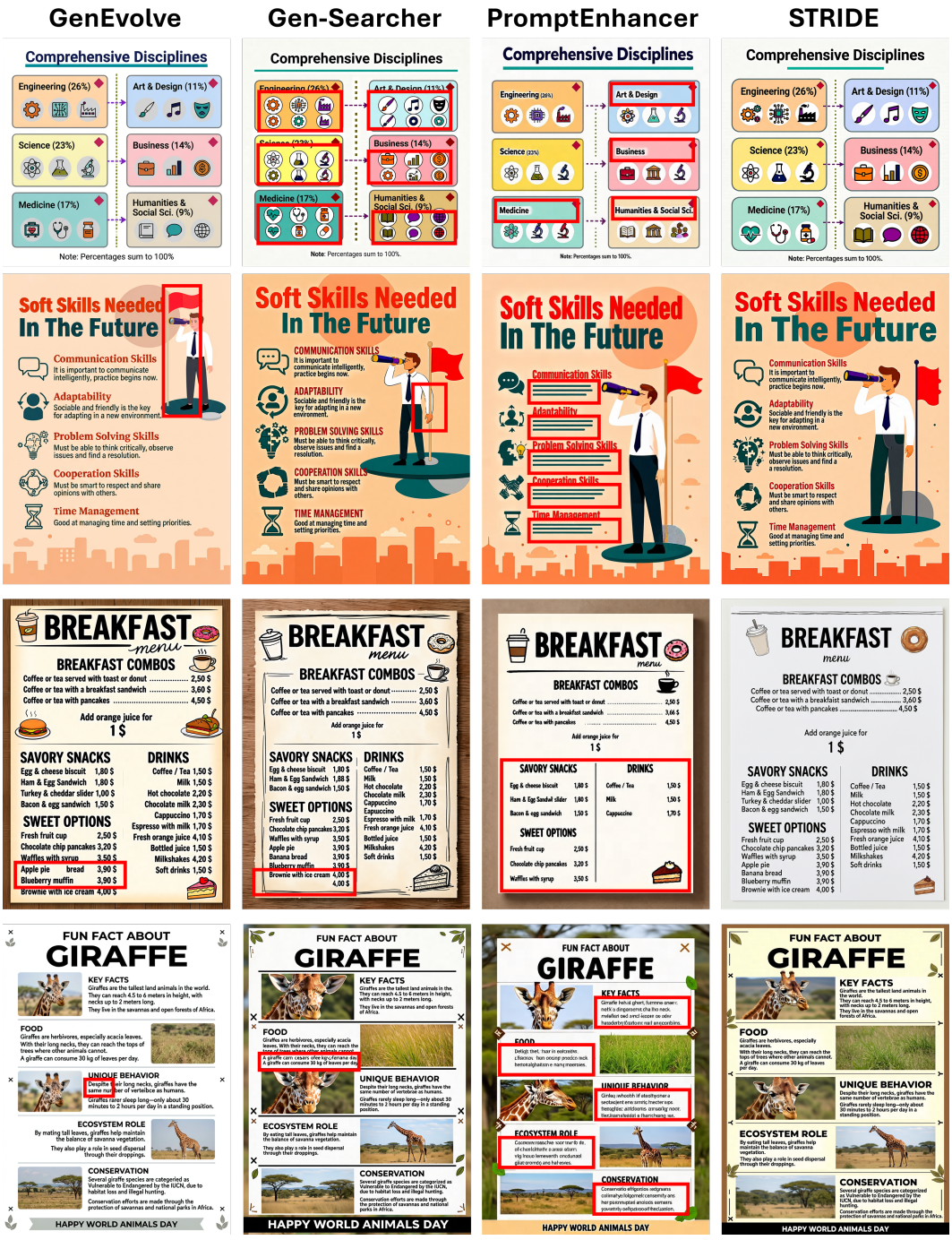

Figure 5: More qualitative comparison with existing agentic methods on the infographic task.

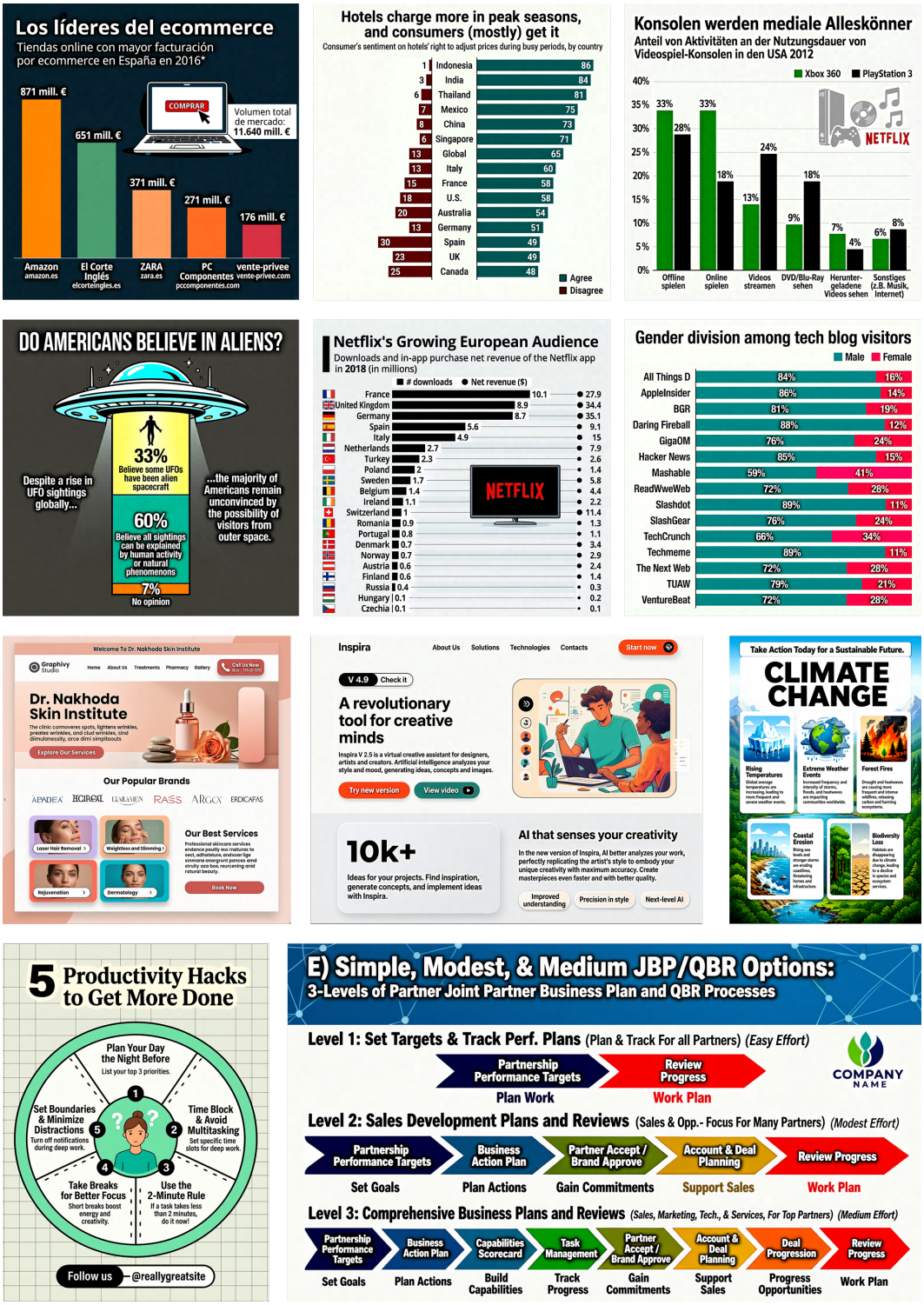

Figure 6: More qualitative results of STRIDE.



Figure 7: Qualitative comparison with existing agentic methods on general WISE task.

H CASE STUDY OF STRIDE

To illustrate the execution process of STRIDE, we present two representative cases in Figures 8 and 9. Each figure shows the initial champion, the perceptor's visual grounding, and the refined result, where green and red boxes indicate correctly and incorrectly satisfied regions, respectively. We additionally report the lead editor's intermediate analysis to make the diagnosis and route-selection process explicit.

Case 1: Localized text repair. In Figure 8, the initial champion satisfies most of the requirements and has a coherent global layout. The perceptor identifies the only remaining failure as a garbled text block, while reporting healthy layout coverage and no region overlap. Based on this evidence, the lead editor correctly classifies the failure as a localized text-rendering error and selects the surgical editing route rather than global redraw. The retoucher therefore modifies only the erroneous text region. The refined result resolves the detected text error while preserving the surrounding layout and already-correct content, demonstrating that STRIDE can perform conservative, targeted refinement when the global composition is already valid.

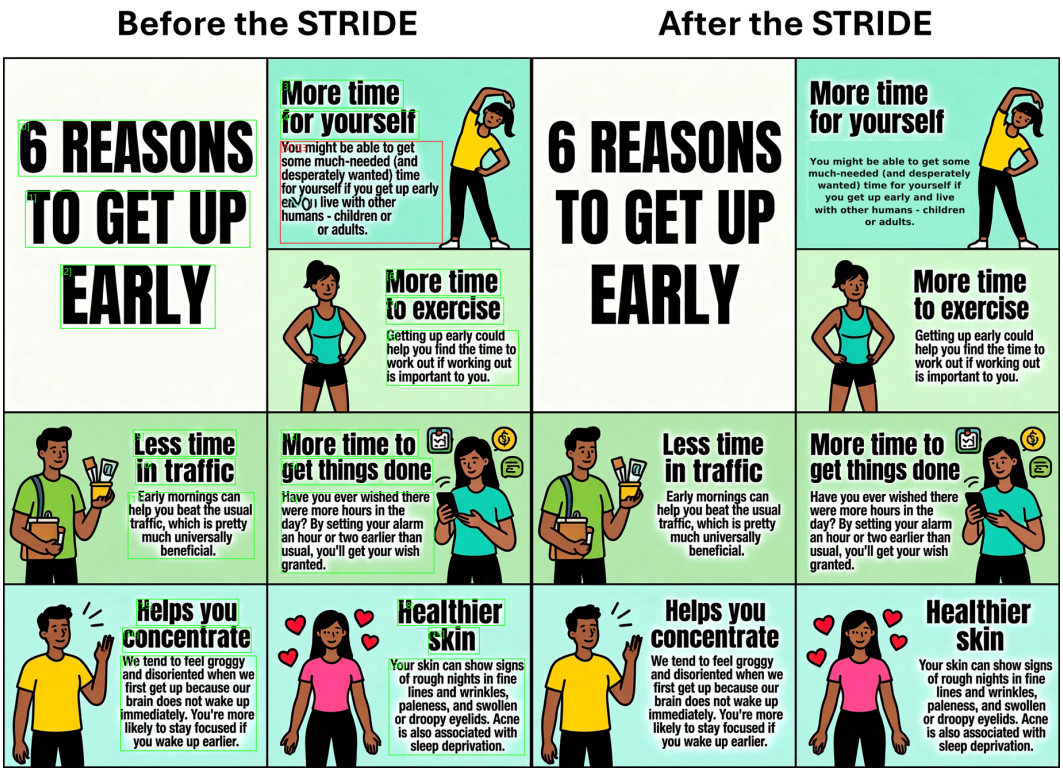

Figure 8: Case study 1: Localized text repair with STRIDE. Guided by visual grounding, the lead editor selects local editing to correct a garbled text block while preserving the surrounding content and layout. Green and red boxes mark regions identified as correct and erroneous, respectively.

Lead Editor Analysis Result

```
{
    'reasoning': 'The subject appears in OCR blocks (garbled block).
Text problem -> surgical. High accuracy, healthy
    layout metrics. Fix the one garbled block.',
    'ok': True,
    'kind': 'analysis',
    'reason': 'agent: The only failure is Q3, whose paragraph is
present in the OCR blocks but garbled, while layout
```

```
    metrics (coverage 38.6%, overlap 0%, hard_acc 1.00) are healthy,
so an in-place text fix suffices.',
    'suggest': 'surgical',
    'model': 'kimi-k3',
    'failure_mode': 'garbled_text_block',
    'confidence': 0.9
}
```

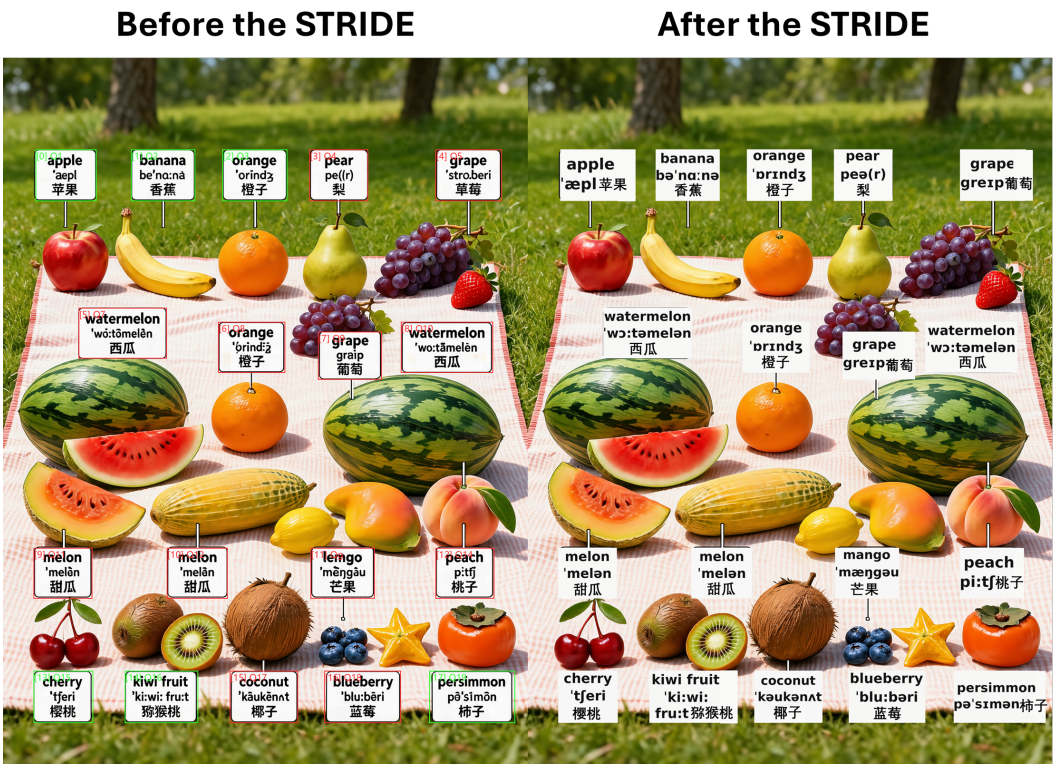

Figure 9: Case study 2: Multiple text corrections with STRIDE. Despite low initial verification scores, the lead editor identifies localized label errors and selects local editing to correct phonetic transcriptions and a Chinese character while preserving the overall layout. Green and red boxes mark regions identified as correct and erroneous, respectively.

Case 2: Multiple localized text errors. Figure 9 presents a more challenging initial champion with a low verification accuracy of 0.10. Although nine hard and nine easy questions initially fail, the perceptor finds that all relevant subjects are present in the OCR blocks and that the layout has no region overlap. The main errors are concentrated in garbled phonetic lines and an incorrect Chinese glyph, rather than missing objects or structural layout failures. The lead editor consequently selects local text repair instead of redrawing the entire image. After refinement, the erroneous labels are corrected while the overall arrangement and visual structure remain intact. This case shows that STRIDE does not equate a low score with a global failure; it uses grounded evidence to distinguish localized rendering errors from structural errors and to select the least disruptive effective action.

Together, these cases demonstrate the reliability of STRIDE's refinement loop: the perceptor turns abstract verification failures into spatially actionable evidence, the lead editor uses this evidence together with the agent state to select an appropriate editing scale, and the retoucher performs targeted corrections while preserving valid content.

Lead Editor Analysis Result

```
{
    'reasoning': 'Let me analyze this carefully. Current state:
accuracy=0.10, hard_acc=0.10, easy_acc=0.10;
    n_failed_hard=9, n_failed_easy=9; OCR has 18 blocks, 12 garbled;
box_coverage=17.1%; box_overlap_rate=0.0%;
    champion_changed_since_last_round=true; first attempt. All
failed subjects appear in OCR text blocks, but many are
    garbled, especially the phonetic second line. This makes the
issue surgical text repair rather than redraw..... .',
    'ok': True,
    'kind': 'analysis',
    'reason': "agent: Layout is intact (0% overlap, all 18 fruit
labels present in OCR) and every failed question's
    subject is visible but has a wrong IPA middle line or one wrong
Chinese glyph, so this is in-place text correction,
    not a redraw.",
    'suggest': 'surgical',
    'model': 'kimi-k3',
    'failure_mode': 'garbled_ipa_label_text',
    'confidence': 0.78
}
```

I Limitations and Future Work

STRIDE has two limitations that suggest directions for future work. First, although the open-weight configuration achieves competitive results, refinement remains dependent on the reliability of rubric verification and spatial grounding. Monotonic admission applies to the internal verifier score and does not guarantee improvement under independent evaluation. Our diagnostic experiments indicate room for improvement in these components, which the modular framework can readily accommodate. Better-calibrated verification and more precise grounding could further strengthen candidate selection and targeted correction.

Second, the current retoucher focuses on text and formula replacement within localized regions, enabling targeted corrections while preserving the surrounding content. Changes to chart geometry or spatial relationships currently rely on the composer's global regeneration. Extending local refinement to structured visual elements would allow a broader range of errors to be corrected without redrawing the entire image. A promising direction is to jointly edit labels and their associated visual marks, further extending the benefits of scale-aware refinement.

J Prompts for Agents in STRIDE

To facilitate reproduction and inspire future research, this section reports the prompts used by the agents in STRIDE. Variables enclosed by braces are filled at inference time.

J.1 Verifier

Verifier System Prompt

```
You are a precise, strict visual checklist judge.
```

Verifier Prompt

```
Evaluate the image against EACH checklist question below. For each
question answer strictly true (the image fully satisfies it) or
false (it does not, or only partially). Be strict: partial credit =
false.
```

```
=== CHECKLIST ({n_questions} questions) ===
{qlines}

Return STRICT JSON only: {{"results":[{{"qid":1,
                "result":true,
                "reason":"..."}},...]}} covering ALL questions, no
extra text.
```

J.2 PERCEPTOR

Perceptor Prompt

```
You are an OCR + layout localizer. List EVERY visible text block in
this image|title, labels, values, formulas, captions, everything.
Be thorough: scan the entire image systematically top->bottom,
left->right.

For each text block give:
1) literal text (verbatim transcription, even if garbled/misspelled)
2) bounding box [x1,y1,x2,y2] as INTEGERS in 0-1000 range
(x=horizontal left->right, y=vertical top->bottom), fit TIGHTLY to
the text
3) garbled flag: true if text has spelling errors, nonsense
characters, broken formulas, or looks corrupted.

IMPORTANT: Do NOT skip any text. List at least {min_blocks}-30
blocks for a typical infographic.

Return STRICT JSON: {{"blocks":[{{"text":"...",
                "box":[x1,y1,x2,y2],
                "garbled":true/false}},...]}} with NO extra
commentary.
```

J.3 LEAD EDITOR

Lead Editor Prompt

```
Decide how to repair a generated infographic that failed part of
its checklist, and write concrete guidance for whichever repair
actor you choose.

=== CURRENT STATE ===
accuracy={accuracy:.2f}  hard_acc={hard_acc:.2f}
easy_acc={easy_acc:.2f}
n_failed_hard={n_fail_hard}  n_failed_easy={n_fail_easy}

=== OCR VIEW (of the CURRENT champion image) ===
{ocr_view}

=== OCR TEXT BLOCKS {ocr_blocks}

=== FAILED QUESTIONS ===
{failed_questions}

=== PRIOR ATTEMPTS ON THIS ITEM ===
{attempt_history}

=== HISTORICAL PRIORS (guidance, not rules | you may overrule them
with reason) ===
{rules_prior}
```

```
 Two strategies are available:
 surgical | OCR the image, compute per-block text corrections,
occlude and re-render those blocks in place. Preserves everything
already correct. Cheap. Cannot fix layout, composition, or missing
visual elements. The renderer repaints text INSIDE the existing
bounding box, so it cannot move, resize, add or delete an element.
 redraw   | regenerate the whole image from an enhanced prompt with
several seeds. Can fix structure and layout. Discards everything
currently correct, so it can regress.

 HOW TO READ THE EVIDENCE
 - A failed question whose subject DOES appear in the OCR text
blocks (even garbled) is a TEXT problem -> surgical.
 - A failed question whose subject appears NOWHERE in the OCR text
blocks usually means the element was never drawn -> redraw.
 - Low box_coverage or high box_overlap_rate points at a broken
layout -> redraw.
 - If champion_changed_since_last_round is false, the OCR numbers
are unchanged BY CONSTRUCTION. Do not read "garbled_rate did not
drop" as "surgical failed" in that case; use PRIOR ATTEMPTS instead.
 - "BROKE previously-correct" in the history is the strongest signal
available: the last repair damaged content that was already right.
Say explicitly what to avoid this round.

 YOUR GUIDANCE
 Write `guidance` for the actor you selected. It is injected
verbatim into that actor's prompt next round, so address the actor,
not the user. Be specific and grounded in the history above: name
the blocks, question ids, or constraints involved and say what to
do differently. If there is no prior attempt, give a short plan
instead. Never invent facts that are not in the evidence above.
Keep it under 1000 characters.

 Return STRICT JSON only:
 {{"strategy":"surgical"|"redraw",
       "confidence":0-1,
       "failure_mode":"short_snake_case_label",
       "reason":"one sentence for a human reader",
       "guidance":"concrete instructions for the selected actor"}}
```

J.4 COMPOSER

Composer System Prompt

```
 You are a prompt engineer for a text-to-image model that generates
infographics.

 You are given the ORIGINAL prompt, the FAILED hard checklist
questions (the generated image violated these constraints), and the
verifier's reasons. You rewrite those failures into short,
imperative, unambiguous constraint sentences and APPEND them to the
original prompt.

 You are called in up to two passes:
 Pass 1 | no web evidence. Answer from your own knowledge and state
your confidence honestly. If the constraints depend on facts you
are unsure of, say so (need_search=true) and propose real search
queries.
 Pass 2 | only happens if you asked for a search. You are shown your
own Pass-1 output plus the search results. Revise the constraints
and update confidence. Keeping a low confidence after searching is
a valid and useful answer | do not inflate it just to look decisive.
```

```
Rules:
- NEVER modify, shorten, or reorder the original prompt. Only
append.
- The appended section starts with: "STRICT LAYOUT/COUNT
CONSTRAINTS (must all hold):"
- One numbered line per failed constraint, phrased as a direct
instruction (e.g. "Panel (E) must be positioned directly below
panel (A).", "Draw EXACTLY five panels labeled (A) (B) (C) (D)
(E).", "The legend swatch color for each method must be identical
to its bar color.").
- Keep the appended section under 900 characters. Prioritize
count/position constraints.
- Never invent facts. An honest low confidence is better than a
confident wrong constraint, because a wrong constraint actively
steers the image model away from the right answer.
- Output ONLY the JSON object described in the user message. No
commentary.
```

Composer Prompt

```
ORIGINAL PROMPT (never modify, reorder, or delete any of it |
append only):
{original_prompt}

=== ITEM ===
domain={domain}  dimension={dimension}

=== CURRENT STATE ===
accuracy={accuracy:.2f}  hard_acc={hard_acc:.2f}
easy_acc={easy_acc:.2f}
n_failed_hard={n_fail_hard}

FAILED HARD QUESTIONS (each must become an explicit, checkable
constraint):
{failed_questions}

JUDGE REASONS (why the verifier rejected them | the most specific
evidence you have):
{judge_reasons}

GUIDANCE FROM ANALYZER (based on previous attempts on this image;
"(none)" on first try):
{guidance}

YOUR PRIOR ANALYSIS (empty on the first pass; on the second pass
this is what YOU wrote before seeing any search results | revise
it, do not restate it):
{prior_analysis}

WEB EVIDENCE (empty unless a search was run; never contradict it):
{evidence}

WHAT TO DO
1. Turn each failed hard question into a short, imperative,
unambiguous constraint and put them in `appended`. Prioritize count
and position constraints | those are what the image model most
reliably obeys.
2. Judge your own factual certainty in `confidence` (0-1). This is
about the WORLD KNOWLEDGE the constraints depend on (names, dates,
formulas, quantities), not about your writing.
3. If `confidence` < 0.7, set `need_search` to true and put 1-3
targeted `search_queries` in the output. Write real search queries
```

```
about the missing facts | do NOT just echo the checklist question,
it makes a poor query. Also include the word `need_search` in
`reasoning`.
 4. If WEB EVIDENCE is present, ground the constraints in it and
update `confidence`.

 HARD RULES
 - NEVER modify, shorten, or reorder the ORIGINAL PROMPT. `appended`
is appended to it verbatim.
 - `appended` must start with: "STRICT LAYOUT/COUNT CONSTRAINTS
(must all hold):"
 - One numbered line per constraint. Keep `appended` under 900
characters.
 - Never invent facts to fill a gap. If you do not know, lower
`confidence` and ask for search.

 Return STRICT JSON only:
 {{"reasoning":"what knowledge these constraints depend on and how
sure you are",
       "confidence":0.0-1.0,
       "need_search":true|false,
       "search_queries":["...","..."],
       "appended":"STRICT LAYOUT/COUNT CONSTRAINTS (must all
hold):\n1. ..."}}
```

J.5 RETOUCHER

Retoucher System Prompt

```
 You are an infographic text repair expert with broad knowledge
(chemistry, math, physics, biology, arts, history, astronomy).

 You are given: the failed checklist questions for a generated
infographic, and the full OCR block list of that image. Your job is
to decide, for each repairable block, what the CORRECT text should
be.

 Priority of evidence, highest first:
 1. What the original prompt / checklist question explicitly
requires.
 2. Established world knowledge (formulas, constants, dates, named
entities).
 3. Inference of intent from the garbled string itself.

 Hard rules:
 - Never invent facts to fill a gap. If you cannot reach the
required confidence, SKIP the block.
 - Only bind to block_index values that exist in the provided OCR
list.
 - Keep replacement text about as long as the original; the renderer
must fit it into the original bounding box, and a much longer
string will be shrunk until unreadable.
 - Output strict JSON only, no commentary.
```

Retoucher Prompt

```
 Repair this infographic by answering failed checklist questions
correctly, AND fixing leftover garbled blocks.

 === FAILED QUESTIONS (HARD prioritized, but fix EASY too) ===
 {failed_questions}
```

```
 === OCR TEXT BLOCKS (all blocks, including correct ones for
context) ===
 {ocr_blocks}

 === CURRENT SCORES ===
 accuracy={accuracy:.2f}  n_fail_hard={n_fail_hard}
n_fail_easy={n_fail_easy}

 === GUIDANCE FROM ANALYZER (based on previous attempts on this
image) ===
 {guidance}

 Instructions:
 1. For each failed question: give the correct answer, bind to the
OCR block index to replace.
    {unbound_instruction}
 3. kind: 'formula' (math/chem -> LaTeX like $H_2O_2$, $x^{{2}}$,
$\frac{{1}}{{2}}$) or 'text'|'label'|'title'|'value' (plain).
 4. confidence >= {min_confidence} required, or skip.
 5. Only bind to a block index that appears in the OCR list above.
Never invent an index.
 6. The block indices above are specific to THIS list. Indices from
any earlier round or from the guidance text may refer to a
different OCR pass | always re-locate the block by its text content
in the list above before using an index.
 7. If the guidance says a previous edit broke a previously-correct
answer, do not repeat it: either leave that block alone or choose a
replacement that satisfies both questions.

 Output JSON:
 {{"targets":[{{"qid":<int or null>,
                "block_index":<int>,
                "correct_text":"...",
                "kind":"formula|text|label|title|value",
                "latex_expr":"(if formula, same as correct_text)",
                "is_hard":true|false,
                "confidence":0-1}},...]}}
```