PhotoFlow: Agentic 3D Virtual Photography Missions

Jiarui Guo[1,2] Haojia Wei[1,3] Yiming Zhang[1,4] Yifei Liu[5] Yuning Gong[5]
Hongjie Zhang[5] Xue Yang[1] Zhihang Zhong[1*]
[1]Shanghai Jiao Tong University [2]Northeastern University [3]University of California, Los Angeles
[4]Cornell University [5]Shanghai AI Laboratory
https://visionary-laboratory.github.io/PhotoFlow/

Abstract

Virtual photography asks an agent to enter a prepared 3D scene with no preselected camera pose or reference image, infer a suitable shot from scene information and a language intent, choose executable camera parameters, and render the final photograph. Recent progress in vision-language models makes this kind of spatial agent increasingly plausible, but the task stresses two capabilities that remain hard to evaluate together: complex 3D spatial understanding and abstract aesthetic judgment. We introduce PhotoFlow, a Director-Reviewer-Reflector agent for closed-loop camera search. The Director builds a soft photographic blueprint and proposes diverse candidate cameras; the Reviewer combines rule checks, visual critique, and pairwise incumbent selection; and the Reflector converts failures into region memory, dead-zone suppression, and high-explore relocation. We also introduce VPhotoBench, a benchmark of 47 Creative-Commons-licensed Blender scenes and 141 language-conditioned photography missions spanning subject placement, relational composition, and atmosphere/style. On held-out experiments, PhotoFlow attains the best perceptual-quality composite point estimate and the highest quality-threshold success rate among one-shot prediction, single-chain reflection, anchor-bank selection, and random search under a six-round rendering budget, with rendering and model-call cost, variance, and failure structure reported for every method. A separate human spatial audit confirms that image-level quality and geometric correctness must be scored separately, since a non-trivial fraction of high-quality outputs still violate task geometry. To our knowledge, this is the first work to make language-conditioned virtual photography in arbitrary Blender scenes an executable agent task, and our results show that an LLM-centered spatial agent can already produce strong photographs in a setting designed to challenge both 3D reasoning and aesthetic choice.

1 Introduction

Virtual photography builds on automated camera control and virtual camera planning, where a system must choose concrete camera specifications to communicate a scene through composition and viewpoint [1, 2, 8, 11, 15]. We study the language-conditioned version: given a controllable 3D scene and a photography intent, a spatial agent must produce a final still image by choosing an executable camera state. Unlike image generation, the output camera pose, look-at target, lens, aperture, and aspect ratio must correspond to a rerenderable view of the scene. The task therefore joins two requirements that are usually evaluated separately: the agent must understand 3D layout and visibility, and the rendered image must satisfy an abstract photographic goal such as subject emphasis, relational composition, or atmosphere.

*Corresponding author. Email: zhongzhihang@sjtu.edu.cn

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

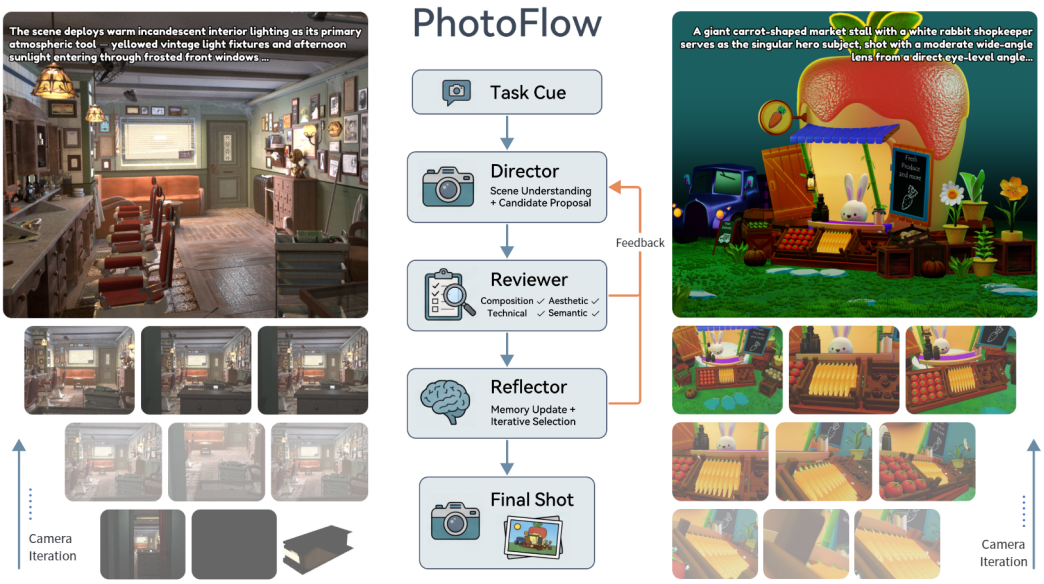

Figure 1: **Virtual photography as spatial-aesthetic decision making.** Given a controllable 3D scene and a language instruction, the agent must choose an executable camera state that satisfies spatial constraints, semantic intent, and photographic quality. The benchmark evaluates the final rendered image together with the search process that produced it.

This combination exposes a difficult gap in current multimodal intelligence. Vision-language models remain unreliable on spatial relations, object orientation, relative depth, and multi-view perception, even in controlled benchmarks with visible objects [14, 17, 20, 27, 30]. Aesthetic evaluation is also not a settled oracle: image-aesthetic and perceptual-quality models are useful proxies, but human preference is subjective and depends on both image attributes and viewer factors [9, 22, 28, 33]. Virtual photography stresses both sides at once because the agent must search through physically valid 3D views while optimizing for a high-level visual intent.

No existing benchmark directly covers this setting: robotic photography emphasizes physical capture, drone cinematography smooth trajectories, aesthetic assessment completed images, embodied navigation paths, and text-to-image generation needs no valid camera state. To our knowledge, this is therefore the first work to study language-conditioned still photography in arbitrary virtual scenes as an executable agent task. Because no baseline suite exists for this exact problem, we construct controlled baselines—one-shot prediction, single-chain reflection, anchor-bank selection, and random search—and use them to identify which failures appear and which agentic mechanisms mitigate them.

We introduce PhotoFlow, a Director-Reviewer-Reflector agent that treats photography as finite-horizon feedback-driven search (Figure 1). The Director proposes diverse candidates from scene scouts, a soft photographic blueprint, global anchors, and region memory; the Reviewer diagnoses rendered previews with rule-based and visual criteria; and the Reflector converts failures into search bias, dead-region suppression, and high-exploration relocation. We also introduce VPhotoBench, a 141-mission benchmark over 47 Creative-Commons-licensed Blender scenes with heterogeneous per-asset terms. Under a six-round rendering budget, PhotoFlow attains the best perceptual-quality composite point estimate and the highest quality-threshold success rate among the tested baselines, while the experiments report render-availability filtering, ablations, search diagnostics, metric-weight sensitivity, wall-clock and token cost, an independent human spatial audit, and human consistency checks. The loop builds on established generate–evaluate–reflect patterns; the contribution is the task, its executable evaluation, and a domain-specific, diagnosable search realization.

Our contributions are:

- We propose PhotoFlow, a Director-Reviewer-Reflector architecture for continuous camera search with soft blueprints, global anchor banks, region memory, four-dimensional review,

pairwise incumbent selection, dead-zone suppression, forced high-explore relocation, and explicit aspect-ratio reasoning.

- We define VPhotoBench, a 141-mission benchmark over 47 Creative-Commons-licensed Blender scenes that couples scene geometry, natural-language intent, aspect-ratio choices, bootstrap protocols, and structured evaluation constraints.
- We report a held-out comparison with failure accounting, ablations, search diagnostics, weight sensitivity, cost accounting, an independent spatial audit, and human preference checks, so that final claims are tied to external metrics rather than internal reviewer scores alone. We release the agent code, benchmark registry, task specifications, scene/license metadata, complete prompts, and evaluation scripts at `https://github.com/Visionary-Laboratory/PhotoFlow`.

2 Related Work

Automated photography and cinematography. Early automated photography systems treated camera placement as motion control under compositional constraints, from the robot photographer of Byers et al. [8] to LeRoP [18] and reinforcement-learning methods such as AutoPhoto [2], all showing that camera placement can be automated as search. Drone and virtual cinematography systems further optimize subject tracking, smoothness, safety, and shot composition under real-time control constraints [7, 23, 25]. Language-driven systems such as ChatCam and recent film agents broaden the interface to conversational control, script-level planning, or multi-agent previsualization [19, 21, 24, 32]. Our task inherits the need for executable camera states, but differs in its design target: we study still photographic decision making in arbitrary-complexity virtual 3D scenes, where the final image must satisfy language-conditioned subject, relation, style, and aspect-ratio constraints rather than only reach a physical capture pose or produce a smooth trajectory.

Aesthetic assessment and view suggestion. Image aesthetic assessment provides the scoring tools that make automated photography measurable. Classic work studied photographic quality attributes and aesthetic datasets [12, 22]; neural methods such as NIMA predict human aesthetic ratings from images [28]; and Creatism demonstrated an end-to-end deep-learning photographer for professional-style image crops and post-processing [13]. Recent 3D aesthetic-field approaches extend aesthetic prediction into continuous 3D viewpoint spaces [29]. These systems are important evaluators or priors, but they do not define a language-conditioned closed-loop agent that must reason about task constraints, aspect ratio, and iterative failures.

Embodied and virtual-environment benchmarks. Embodied AI benchmarks such as Matterport3D, Gibson, Habitat, and Room-to-Room have made navigation and spatial reasoning reproducible in 3D environments [4, 10, 26, 31]. Their protocols make movement part of the task: navigation reports success with path length or SPL [3], VLN path-fidelity metrics such as nDTW and SDTW reward trajectories that follow the reference route [16], and LLM-based VLN agents such as NavGPT reason over navigation history and explorable directions before choosing the next action [34]. Virtual photography borrows this reproducibility discipline but evaluates a different object: the final camera state and rendered image, not the route by which it was discovered.

3 PhotoFlow

3.1 Task formulation

We define a virtual photography mission as a five-tuple

$$b = (S, x, u, A, E), \tag{1}$$

where S is a controllable Blender scene, x is a natural-language photography instruction, u is the bootstrap information available to the agent, A is the allowed aspect-ratio set, and E is a structured evaluation specification. The specification E is not a restatement of the prompt. It encodes checkable task intent such as primary subject visibility, screen placement, desired subject scale, camera-angle preference, symmetry, depth emphasis, and hard-failure conditions.

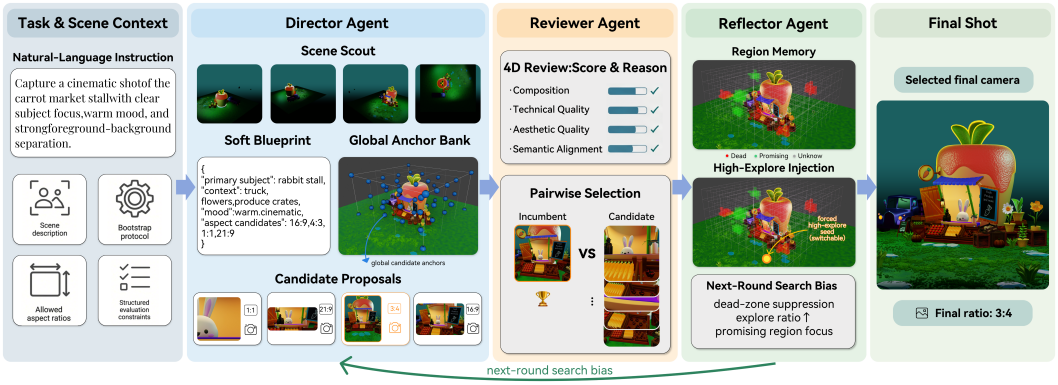

Figure 2: **PhotoFlow pipeline.** The system first scouts the scene and constructs a soft photographic blueprint. The Director proposes candidate cameras from global anchors, region-memory-guided seeds, and a forced high-explore lane. Candidate previews are rendered in parallel, scored by a structured Reviewer, and summarized by a Reflector that updates search bias and forbidden regions for the next round.

The output is an executable camera state

$$c = (p, \ell, f, d, r), \tag{2}$$

where $p \in \mathbb{R}^3$ is the camera position, $\ell \in \mathbb{R}^3$ is the look-at point, f is focal length, d is aperture, and $r \in A$ is the selected aspect ratio. A renderer maps (S, c) to an image $I = \mathcal{R}(S, c)$. This is the key difference from image generation: the final photograph must correspond to a concrete, rerenderable view of the scene. PhotoFlow therefore does not directly regress x to a single c; it performs finite-horizon search over T rounds, rendering candidate views, receiving feedback, and updating its search bias.

3.2 Scouting and blueprint

Directly asking a large model to output continuous camera parameters from a raw object list is unstable. PhotoFlow therefore begins with scene scouting. From Blender, it extracts three kinds of input. The geometric scene summary contains object names, bounding boxes, centers, scene extents, and coarse visibility proxies. The textual topology summary converts these statistics into relations such as dominant objects, foreground/background groups, vertical structure, and likely open regions. The global scout views are low-sample preview renders from a small set of canonical or visibility-oriented cameras around the scene. The extracted scene blueprint is used as a photographic search substrate rather than a pedestrian reachability graph: in virtual production, a visually meaningful camera can be valid even when the set has no realistic entrance or traversable route to that position.

The Director then converts the instruction and scouting evidence into a soft blueprint. This conversion is a text-only LLM parsing step over the instruction, the reference object, and the scene object/topology summaries; scout images are not shown to it, and are supplied instead to the candidate-proposal call, while the scout camera poses seed deterministic relocation anchors. The blueprint call uses a constrained schema: the model identifies the likely primary subject, useful context objects, preferred composition cues, camera-angle preference, camera-zone preference, look-toward target, axis preference, symmetry preference, semantic vibe, and negative preferences. The blueprint is soft because these fields are preferences, not hard constraints: they bias search while allowing multiple valid photographs instead of forcing one template.

3.3 Director

The Director proposes candidates on top of interpretable spatial priors. A global anchor bank is a finite set of coarse camera seeds $\{a_i\}$ defined before local search begins. Each anchor contains an initial camera position, look-at target, approximate lens choice, aspect-ratio hint, and prior score. The bank is assembled from three families: scene-bounding-box cameras along global axes, sides, and a high overview; visibility-grid anchors around the canonical subject and context objects; and

relocation anchors derived from the scout camera poses. Scene-bounding-box targets and blueprint look-toward targets supply look-at points for the first family, so the blueprint influences anchor construction through its target fields rather than as a separate family. Because these anchors are decoupled from the current incumbent, they remain available when the search falls into a locally acceptable but globally weak viewpoint. The bank-entry check only verifies that camera position, look-at target, and focal length are present and finite; it is not a collision, occlusion, visibility, or semantic-validity test (Appendix C).

At each round, the system builds a mixed seed pool before asking the LLM to propose candidates. A seed is a partially specified camera hypothesis, usually derived from the current incumbent, a promising memory region, a global anchor, or a geometry probe. Region memory is produced by the Reflector from previous rounds: each rendered candidate is assigned to a coarse spatial cell and the cell stores visits, scores, failures, and improvement signals. Promising regions receive local refinement seeds; unknown or dead regions increase the share of global anchors and geometry probes. The LLM then turns the seed pool and reviewer feedback into complete candidate proposals

$$y_j = (c_j, \rho_j), \quad c_j = (p_j, \ell_j, f_j, d_j, r_j),$$

where ρ_j is a short rationale used only for interpretation and later reflection. If model output is malformed or underspecified, the implementation falls back to seed candidates and lightweight perturbations so that the loop remains executable.

3.4 Reviewer

The Reviewer is designed to expose why an image fails. The environment first computes rule-based indicators from projection geometry and task constraints: subject visibility, screen placement, and subject scale are estimated by projecting the target object's bounding box into the camera frame, and hard failures mark missing subjects, extreme occlusion, invalid cameras, or gross violations of required view type. A visual reviewer then scores the rendered preview along four dimensions: composition quality, technical quality, aesthetic quality, and semantic alignment. Together with the two rule-side signals, the six Reviewer signals are combined as

$$J(c) = 0.10m_1 + 0.10m_2 + 0.15m_3 + 0.15m_4 + 0.25m_5 + 0.25m_6, \tag{3}$$

where m_1, m_2 are deterministic projection-side signals and $m_3, \ldots, m_6$ are VLM-side image scores. The fixed weights are set before held-out evaluation and used only for internal search. The score is not a final evaluation metric; it ranks candidates within a run, while the dimension-wise reasoning is passed to the Reflector.

The Reviewer also performs pairwise incumbent selection: instead of replacing the best image by scalar score alone, it compares the incumbent and the new candidate directly, identifies the stronger image per dimension, and selects the image that is both better and more stable for further optimization, which reduces oscillation under noisy scalar scores.

We considered preference-based Bayesian optimization (PBO) for this selection step because recent agentic aerial cinematography uses pairwise visual preferences to refine camera poses [19]. That system formulates full pose optimization, although its reported aerial setting optimizes translation and yaw, and it samples many challenger poses per update (e.g., 64 candidates per iteration) with tens to roughly one hundred preference updates before convergence. Such sampling is expensive in virtual photography, where every comparison requires a render, and in our tests frontier vision-language models already produced near-human pairwise comparisons for near-neighbor photographic choices. PhotoFlow therefore uses direct reviewer comparison for the round incumbent instead of a separate PBO loop.

3.5 Reflector

The Reflector turns round-level feedback into future control signals. Continuous space is discretized into cubic region cells with side length $h = \max(0.12\,\mathrm{sceneScale}, 0.9)$. Each region records visit count, best score, semantic score, poor hits, promising hits, improvement hits, and stagnation hits, and is labeled as unknown, promising, or dead. A region becomes promising if its best internal score reaches .68, its best semantic score reaches .70, or it receives a promising hit; it becomes dead after repeated low-score visits or repeated stagnation without improvement. Dead regions are converted into forbidden zones, while promising regions may still be exploited.

To prevent premature local collapse, the architecture includes a forced high-explore lane. In each round, when feasible, one anchor seed a is drawn from the global anchor bank according to a priority score

$$s(a) = \pi(a) + u(a) + \min\left(\frac{\|p_a - p_t\|_2}{2h}, 2.0\right) - 0.35\, n(a) - 0.40\, \mathbf{1}[a \in \text{promising}], \quad (4)$$

where $\pi(a)$ is the anchor prior from scouting, p_a and p_t are the anchor and current-incumbent camera positions, $n(a)$ is the visit count of the anchor's region, and $u(a) = 1.2$ for an unknown region and $.25$ otherwise. Anchors in dead regions are skipped before ranking. This is a structured curiosity channel rather than a random restart: one candidate always probes low-visit, non-dead, spatially meaningful anchors, optionally at a different aspect ratio.

3.6 Rendering and framing

Rendering is the main systems bottleneck in iterative virtual photography. PhotoFlow decouples candidate preview rendering from agent logic by launching external Blender subprocesses when the source scene and binary path permit it, with a serial fallback otherwise. Preview samples are capped at 64 and render settings are restored afterward, so final render quality is not polluted by preview settings. Although previews are parallelizable in principle, our measured replay of fixed candidate batches with 1/2/4 workers produced no process-level speedup (0.35–$0.70\times$) because repeated scene loading dominates, so we report measured rather than hypothetical parallelism; persistent render workers remain engineering work (Appendix G).

Aspect ratio is a compositional decision: candidates must choose a ratio from A and justify it, and after search a final selection step re-examines the best preview using scene axis strength, subject concentration, environmental breadth, and requested atmosphere. The final output is rendered at the resolution implied by the chosen ratio.

3.7 Implementation and configuration

All Director, Reviewer, and Reflector calls, as well as every LLM baseline, use the same recorded multimodal model identifier, `gpt-5.4`, through one agent class and an OpenAI-compatible proxy; the roles are prompt-conditioned instances, not separately trained models. The proxy exposed no immutable upstream snapshot, so we report the recorded identifier and exact request configuration rather than implying stronger provenance. The six role prompts are zero-shot, with decoding parameters left at provider defaults (temperature, top-p, and maximum tokens omitted; no API seed sent; local seed 0 only controls deterministic fallback). Geometry rules, anchor construction, schema validation, fallback generation, and region-memory updates are deterministic code, and the external UniPercept evaluator is independent of the internal Reviewer. Defaults: six rounds, four candidates per round, four scout views, two preview workers, 64 preview samples, 128 final samples, Blender 4.4.3, A800 80 GB. Complete prompts, schemas, and prompt hashes are given in Appendix B.

4 VPhotoBench: Benchmark Formulation

4.1 Benchmark composition

VPhotoBench instantiates the task formulation from Section 3.1 over 47 Creative-Commons-licensed Blender scenes with heterogeneous per-asset terms (CC-BY 22, CC-BY-SA 12, CC0 12, CC-BY-ND 1). 30 scenes come from the official Blender Demo Files archive [6], and 17 come from Blend Swap [5]. Each scene is paired with three natural-language missions—subject placement, relational composition, and atmosphere/style—yielding 141 runnable task instances. Table 6 in Appendix D reports the scene distribution over visual style, environment, and subject type. Each scene also carries a five-level complexity label assigned with a four-dimensional rubric (geometric density, material complexity, lighting complexity, and compositional richness) whose levels and frozen distribution $\{1{:}1, 2{:}10, 3{:}14, 4{:}16, 5{:}6\}$ (mean 3.34) are documented in Appendix D. Independent per-rater records from the construction stage were not retained, so no original-stage agreement statistic exists; a retrospective two-rater re-annotation of all 47 scenes reproduces the labels exactly on 43/47 scenes (within one level on 47/47; unweighted $\kappa = .884$, quadratic-weighted $\kappa = .959$, Spearman $\rho = .964$). The release package includes the scene registry, task JSON files, evaluation

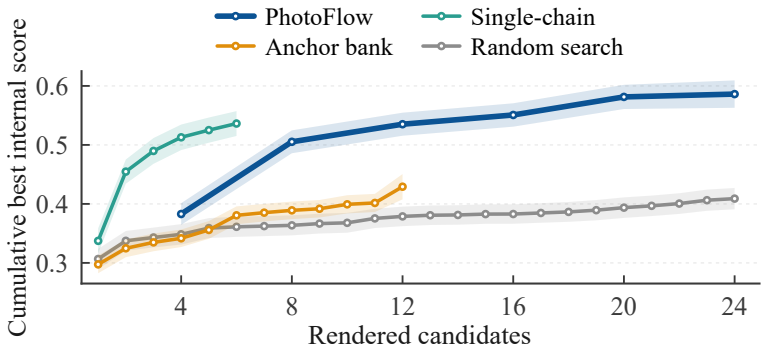

Figure 3: **Search diagnostic.** Internal cumulative best score vs. rendered candidates, aligning PhotoFlow's four-preview rounds with the per-candidate baselines; bands are ± 1 s.e. and curves stop where fewer than 15 runs remain. Process diagnostic only; external metrics are the main evidence (Appendix I).

specifications, per-scene source/license metadata, and the annotation rubric; original assets remain governed by their upstream licenses.

5 Experiments

We evaluate PhotoFlow by separating two questions: whether the benchmark exposes spatial-aesthetic failures that are invisible to single-score evaluation, and whether a closed-loop Director-Reviewer-Reflector search improves camera selection under a fixed rendering budget. Because the agent uses an internal Reviewer during optimization, final comparisons are based on external image metrics, an independent human spatial audit, and human consistency checks rather than internal scores alone.

5.1 Protocol

The main benchmark unit is $(\text{scene}, \text{instruction})$. We use 24 development missions for prompt and threshold selection and reserve 117 missions for held-out testing. Each method is launched on the same 117 held-out missions with matched final render settings, external evaluators, and random seeds. For image-quality means, we apply a task-level common-completed rule: a task is included only if every compared method produces a final image and external scores; otherwise it remains in the failure log. This leaves 90 common completed tasks and excludes the same 27 task IDs for every method because their scenes triggered systems-level render failures under large-scale multi-method evaluation (21 no-first-image timeouts, 3 no-final-image events, and 3 Blender crashes per method). The rule is method-independent rather than winner-selective: retained tasks preserve an exactly balanced mission split of 30 subject-placement, 30 relational-composition, and 30 atmosphere/style missions, and cover all scene families and complexity levels 2–5. The filter is necessary because some authored scenes need hours for one final image, but it does not change the 141-task benchmark definition.

For iterative methods, the main protocol uses a maximum budget of $T = 6$ rounds and four preview candidates per round for PhotoFlow; completed runs average 20.8 preview renders because early stopping, malformed proposals, retries, or unavailable preview slots can reduce the realized count. The production loop also stops when the round-over-round best score plateaus, so the 90 main tasks executed 3/4/5/6 rounds for 3/21/21/45 tasks respectively; we therefore report as-executed cost rather than full six-round cost. Random Search evaluates 24 independent candidate views, Iterative Single-Chain Reflection evaluates one preview per round for six rounds, Anchor Bank Best-of-N scans the generated scout/anchor bank (mean 12.6 anchors, median 11), and Single-Step LLM renders one final prediction. These controls isolate sources of performance rather than equalize compute, so we report preview, model-call, and wall-clock costs alongside quality, and all final images are scored only after the search procedure completes.

The primary comparison includes Single-Step LLM, Iterative Single-Chain Reflection, Anchor Bank Best-of-N, Random Search, and PhotoFlow. Anchor Bank Best-of-N is a critical strong baseline: it

Table 1: **Main comparison on 90 common completed held-out tasks.** M_{pq} is the external perceptual-quality composite in Eq. 5. $\mathrm{Succ@0.55}$ is the fraction of tasks with $M_{\mathrm{pq}} \geq 0.55$.

Method	$M_{\mathrm{pq}} \uparrow$	Succ@0.55 $\uparrow$	IAA $\uparrow$	IQA $\uparrow$	ISTA $\uparrow$
Single-Step LLM	$.514 \pm .130$	.400	.447	.470	.603
Anchor Bank Best-of-N	$.519 \pm .110$	.378	.464	.481	.593
Random Search	$.527 \pm .124$	.400	.483	.492	.589
Single-Chain Reflection	$.567 \pm .115$	.567	.530	.545	**.616**
PhotoFlow	$\mathbf{.578 \pm .120}$	**.622**	**.550**	**.564**	.614

uses the same scout and anchor bank as PhotoFlow but removes reflection and cross-round memory, testing whether gains come from the agent loop rather than good initial anchors alone.

Baseline scope. VLN was the closest embodied formulation we considered because it also studies language-conditioned decisions in 3D environments, and comparing against it clarified the boundary of the new task (Appendix Figure 4). VLN assumes a navigable graph and scores movement policies, where the stop position and the path both contribute to success [3, 4, 16]; virtual photography scores the final camera state, and the path used to discover it does not affect the rendered image. This distinction matters in authored scenes arranged for appearance rather than traversability, so we compare against baselines that optimize final camera selection directly.

Metrics. We use external metrics for all main comparisons. UniPercept [9] provides three image-side perceptual scores: image aesthetic assessment M_{iaa}, image quality assessment M_{iqa}, and image structure-and-texture richness M_{ista}. ISTA measures structural and textural richness rather than alignment with the instruction, so the composite below is an author-defined perceptual-quality score:

$$M_{\mathrm{pq}} = \omega_a M_{\mathrm{iaa}} + \omega_q M_{\mathrm{iqa}} + \omega_s M_{\mathrm{ista}}, \quad (\omega_a, \omega_q, \omega_s) = (0.40, 0.20, 0.40). \tag{5}$$

We weight aesthetics and ISTA more heavily because a virtual photograph must both look strong and preserve visual structure and style; technical quality is included with lower weight because all methods use the same renderer. None of these three image-only scores receives the task text, so none directly measures instruction alignment or geometric correctness; both are evaluated separately below and in Appendix F. We additionally report $\mathrm{Succ@0.55}$, the fraction of tasks with $M_{\mathrm{pq}} \geq 0.55$, as a quality-threshold success measure rather than an independent task-success metric. Because the composite is linear, we enumerate all 5,151 non-negative weight triples at .01 resolution: PhotoFlow ranks strictly first in 5,085 of them (98.7%), including equal weights (.5759 vs. .5636 for Single-Chain), and the 66 exceptions lie in the ISTA-dominant corner ($\omega_s \geq .90$) (Appendix E). Structured constraint logs and hard-failure tags are retained for failure accounting and qualitative diagnosis; we keep them as diagnostic columns rather than main ranking columns because the export is log-based rather than a Blender ray-cast recomputation, which makes it less architecture-neutral than the external image metrics.

Table 1 shows that the closed-loop agent gives the best point estimate of the external perceptual-quality composite under a six-round budget. The gap is largest over one-shot and anchor-only policies: PhotoFlow wins 68/90 paired tasks against Anchor Bank Best-of-N and 60/90 against Random Search. The strongest baseline is Iterative Single-Chain Reflection, which slightly leads ISTA but trails PhotoFlow on the composite, quality-threshold success, aesthetics, and image quality. That comparison is close and we interpret it as a modest gain: $+.0108$ absolute ($+1.9\%$ relative) and $+5.6$ percentage points of quality-threshold success, with 49/90 paired wins (sign test $p = .46$), 12 versus 7 discordant quality-threshold successes (McNemar $p = .36$), and a paired bootstrap 95% interval of $[-.0074, .0288]$ that crosses zero. Eight of the 12 additional successes are spatial missions, where structured relocation helps most, and PhotoFlow is first in every mission category (Appendix H). The gain over Single-Chain is thus a point-estimate and reliability effect under unequal compute.

Cost. The additional search is expensive and we report it explicitly. The 90-task logs average 20.8 preview renders, 48.8 model calls, and 160k tokens for PhotoFlow, versus 6.0, 19.0, and 81k for Single-Chain and 1.0, 1.0, and 6k for Single-Step. An outer wall-clock measurement on three preselected EEVEE subject-placement scenes (complexities 2/3/5, one run each, $n = 3$; Blender 4.4.3, A800 80 GB) gives mean/median end-to-end times of $890.9/795.7$ s, $199.6/203.7$ s, and $43.7/52.4$ s

Table 2: **Ablation study and search diagnostics.** Image metrics are external scores on the same 90 common completed tasks. Coverage, Collapse, and Revisit are log-based search diagnostics computed from candidate region keys.

Variant	$M_{\mathrm{pq}} \uparrow$	Succ@0.55 $\uparrow$	IAA $\uparrow$	IQA $\uparrow$	ISTA $\uparrow$	Coverage $\uparrow$	Collapse $\downarrow$	Revisit $\downarrow$
Full PhotoFlow	.578	**.622**	.550	.564	.614	**.453**	**.389**	**.547**
w/o region memory	.572	.544	.539	.559	.611	.356	.411	.644
w/o high-explore	**.584**	.611	**.557**	**.576**	**.616**	.331	.456	.669

respectively (Table 8 in Appendix G). PhotoFlow is thus $4.46\times$ slower than Single-Chain and $20.4\times$ slower than Single-Step in mean wall-clock time. With $n = 3$ we report these figures as a cost accounting rather than a significance claim.

5.2 Ablations

The ablation study is organized around the three-role design. The Director cannot be removed wholesale because every runnable method must still propose executable camera poses; instead, its multi-candidate behavior is tested by the single-chain baseline, which removes parallel preview rendering and keeps only one feedback trajectory. The Reviewer role is tested by the anchor-bank best-of-N baseline, which removes structured review and reflection while preserving the same initial anchor pool. Table 2 therefore focuses on the remaining removable mechanisms: region memory for the Reflector, and high-explore relocation as the special safeguard against local collapse. In addition to external quality, we report raw-log search diagnostics: region coverage, local collapse, and revisit rate. These diagnostics are not geometric correctness metrics; they explain whether a variant searches too narrowly, revisits low-value regions, or loses exploration diversity.

Table 2 shows that the component effects are not purely monotonic, which is expected for a finite-budget search agent. Region memory provides the cleanest support for the Reflector design: removing it lowers external quality and quality-threshold success while increasing revisits. High-explore is a safeguard rather than a universally beneficial proposal source: disabling it raises some external averages on this subset, but also substantially lowers coverage and increases collapse/revisit rates, matching its intended role. A case study in Appendix Figure 5 shows the full system reaching $M_{\mathrm{pq}} = .696$ where a variant without high-explore stays at .527; because a forced lane consumes one candidate slot, we expose it as a switchable component.

5.3 Spatial correctness audit

Because M_{pq} is image-level, we add a method-independent spatial audit on the 59 valid subject-placement and relational-composition missions. Three task-specific constraints were frozen from each instruction before rating; two human raters labeled all $59 \times 5 = 295$ anonymous final images (1,770 atomic judgments) as Yes/No/Unclear without knowing the method. Raw inter-rater agreement is 70.4% (Cohen's $\kappa = .486$; nominal Krippendorff's $\alpha = .482$). With Yes$= 1$ and No/Unclear$= 0$, mean constraint-level satisfaction and all-three-constraint success are reported in Table 3. Both iterative methods are far above the three non-iterative baselines, but PhotoFlow is not better than Single-Chain on this axis: the paired differences are $-.011$ (95% CI $[-.124, .102]$) for rater A and $-.051$ ($[-.124, .023]$) for rater B, while all-three success is higher for PhotoFlow under rater A ($.508$ vs. $.441$) and lower under rater B ($.441$ vs. $.458$). On this axis, PhotoFlow therefore does not show a consistent advantage over Single-Chain. Across the 295 images, M_{pq} correlates only moderately with human constraint satisfaction (Pearson $.468$/ $.535$, Spearman $.486$/ $.545$), and 36.1–38.9% of PhotoFlow images with $M_{\mathrm{pq}} \geq .55$ still receive at least one explicit spatial No (45.6–59.6% across all methods). Image quality and geometric correctness are therefore related but not interchangeable, and we report them as separate axes.

Failure analysis. We complement the two qualitative failures with a systematic count over the 34 PhotoFlow tasks below the quality threshold on the 90-task set. Two annotators assigned one primary and up to two secondary tags; agreement on a unique primary cause was only 11.8%, so we report non-exclusive tag-presence ranges instead of a single-cause distribution. Low perceptual quality (dark, noisy, or flat renders) appears in 14–21 of 34 cases, wrong viewpoint or violated spatial relation

Table 3: **Independent human spatial audit** on 59 spatial missions ($\times 5$ methods $= 295$ anonymous images, 1,770 judgments). Yes$= 1$, No/Unclear$= 0$. "All three" is the fraction of images satisfying every frozen constraint.

Method	Rater A mean	Rater A all three	Rater B mean	Rater B all three
Single-Step LLM	.384	.169	.384	.237
Anchor Bank Best-of-N	.215	.068	.271	.136
Random Search	.192	.068	.260	.169
Single-Chain Reflection	**.672**	.441	**.655**	**.458**
PhotoFlow	.661	**.508**	.605	.441

in 11–19, style/atmosphere mismatch in 7–16, absent or severely occluded subject in 6–7, search stagnation or redundant regions in 6–7, and wrong placement/scale/framing in 4–6 (Appendix H). The dominant families are thus perceptual quality and viewpoint/spatial-relation errors, both of which the current Reviewer observes only indirectly through rendered previews.

5.4 Human consistency

Because visual quality cannot be fully reduced to automatic metrics, we run a two-stage human subset study. After participant quality control, 30 of 31 participants contribute 780 valid response rows: 450 multi-image preference responses and 300 PhotoFlow-only Likert ratings. Stage 1 collects aesthetic and alignment preferences across methods; Stage 2 collects mean-opinion scores on PhotoFlow outputs and correlates them with automatic metrics. The study is small, serves as a consistency check rather than population-level validation, and does not measure spatial correctness.

PhotoFlow receives the highest selection rate in both aesthetic and alignment choices (31.33% and 30.22%, with 95% intervals .271–.356 and .260–.347), with Iterative Single-Chain Reflection as the closest non-ours method, and M_{pq} correlates strongly with human MOS mean on the 24 rated PhotoFlow outputs (Pearson .827, Spearman .697; Appendix Table 10). These rates indicate a consistent but not overwhelming preference, and they support M_{pq} as a practical image-quality proxy while leaving room for human disagreement; Appendix Figures 6 and 7 show qualitative cases in a prompt/preview/final layout.

6 Limitations

PhotoFlow depends on assumptions we test rather than hide. The anchor bank bounds exploration: if scouting misses a region the high-explore lane stays weak, and bank-entry validation checks only finiteness, not collision or semantics. The Reviewer is a noisy proxy (internal vs. external quality: Pearson .481, Spearman .419), so we use external evaluators and human checks, and since the composite is image-level, high scores can still violate task geometry. The gain over the strongest baseline is modest, with a zero-crossing interval and $4.46\times$ the wall-clock cost of Single-Chain; without threshold-sensitivity or adapted PBO baselines, conclusions stay scoped to this benchmark. The main table is a common-completed comparison, not an availability score, and released failure logs make the accounting auditable; physical or dynamic scenes would add control, collision, and temporal constraints.

7 Conclusion

We presented VPhotoBench and PhotoFlow, a benchmark and an agent for language-conditioned virtual photography. The benchmark turns aesthetic camera selection into a reproducible protocol, and the agent turns photography into closed-loop search over executable camera states. The claim is narrow: virtual photography needs joint evaluation of spatial constraints and aesthetic intent, and camera-search agents need structured feedback to escape poor local viewpoints. Our measurements delimit it: the gain over the strongest baseline is a modest point estimate at higher compute, and image-level quality does not certify geometric correctness. Reporting both axes with cost and failure structure is a concrete path toward measuring spatial-aesthetic intelligence in controllable 3D worlds.

Acknowledgments and Disclosure of Funding

Competing interests. The authors declare no competing financial interests related to this work. **Data and assets.** VPhotoBench uses Creative-Commons-licensed Blender scenes with heterogeneous per-asset terms; per-scene source, author, license, and verification date are released with the metadata, and original assets remain under their upstream licenses. **Human study.** The preference study collected only anonymous ratings of rendered images, with no personal or sensitive data.

References

[1] Rafid Abdullah, Marc Christie, Guy Schofield, Christophe Lino, and Patrick Olivier. Advanced composition in virtual camera control. In *Smart Graphics*, volume 6815 of *Lecture Notes in Computer Science*, pages 13–24, 2011. doi: 10.1007/978-3-642-22571-0_2.

[2] Hadi AlZayer, Hubert Lin, and Kavita Bala. AutoPhoto: Aesthetic photo capture using reinforcement learning. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 944–951, 2021. doi: 10.1109/IROS51168.2021.9636788.

[3] Peter Anderson, Angel X. Chang, Devendra Singh Chaplot, Alexey Dosovitskiy, Saurabh Gupta, Vladlen Koltun, Jana Kosecka, Jitendra Malik, Roozbeh Mottaghi, Manolis Savva, and Amir R. Zamir. On evaluation of embodied navigation agents, 2018. arXiv:1807.06757.

[4] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3674–3683, 2018.

[5] Blend Swap. Blend swap. `https://www.blendswap.com/`, 2025. Accessed 2026-11-02.

[6] Blender Foundation. Blender demo files. `https://www.blender.org/download/demo-files/`, 2025. Accessed 2026-11-02.

[7] Rogerio Bonatti, Wenshan Wang, Cherie Ho, Aayush Ahuja, Mirko Gschwindt, Efe Camci, Erdal Kayacan, Sanjiban Choudhury, and Sebastian Scherer. Autonomous aerial cinematography in unstructured environments with learned artistic decision-making. *Journal of Field Robotics*, 37(4):606–641, 2020. doi: 10.1002/rob.21931.

[8] Zachary Byers, Michael Dixon, Kevin Goodier, Cindy M. Grimm, and William D. Smart. An autonomous robot photographer. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2636–2641, 2003. doi: 10.1109/IROS.2003.1249268.

[9] Shuo Cao, Jiayang Li, Xiaohui Li, Yuandong Pu, Kaiwen Zhu, Yuanting Gao, Siqi Luo, Yi Xin, Qi Qin, Yu Zhou, Xiangyu Chen, Wenlong Zhang, Bin Fu, Yu Qiao, and Yihao Liu. UniPercept: Towards unified perceptual-level image understanding across aesthetics, quality, structure, and texture, 2025. arXiv:2512.21675.

[10] Angel X. Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Nießner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3D: Learning from RGB-D data in indoor environments. In *International Conference on 3D Vision*, pages 667–676, 2017. doi: 10.1109/3DV.2017.00081.

[11] Marc Christie, Rumesh Machap, Jean-Marie Normand, Patrick Olivier, and Jonathan Pickering. Virtual camera planning: A survey. In *Smart Graphics*, volume 3638 of *Lecture Notes in Computer Science*, pages 40–52. Springer, 2005. doi: 10.1007/11536482_4.

[12] Ritendra Datta, Dhiraj Joshi, Jia Li, and James Z. Wang. Studying aesthetics in photographic images using a computational approach. In *European Conference on Computer Vision*, pages 288–301, 2006. doi: 10.1007/11744078_23.

[13] Hui Fang and Meng Zhang. Creatism: A deep-learning photographer capable of creating professional work. *arXiv preprint arXiv:1707.03491*, 2017.

[14] Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. BLINK: Multimodal large language models can see but not perceive. In *European Conference on Computer Vision*, 2024.

[15] Li-Wei He, Michael F. Cohen, and David H. Salesin. The virtual cinematographer: A paradigm for automatic real-time camera control and directing. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, pages 217–224, 1996. doi: 10.1145/237170.237259.

[16] Vihan Jain, Gabriel Magalhaes, Alexander Ku, Ashish Vaswani, Eugene Ie, and Jason Baldridge. Stay on the path: Instruction fidelity in vision-and-language navigation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 1862–1872, 2019. doi: 10.18653/v1/P19-1181.

[17] Amita Kamath, Jack Hessel, and Kai-Wei Chang. What's "up" with vision-language models? investigating their struggle with spatial reasoning, 2023. arXiv:2310.19785.

[18] Hao Kang, Jianming Zhang, Haoxiang Li, Zhe Lin, TJ Rhodes, and Bedrich Benes. LeRoP: A learning-based modular robot photography framework, 2019. arXiv:1911.12470.

[19] Yifan Lin, Sophie Ziyu Liu, Ran Qi, George Z. Xue, Xinping Song, Chao Qin, and Hugh H.-T. Liu. Agentic aerial cinematography: From dialogue cues to cinematic trajectories, 2025. arXiv:2509.16176.

[20] Fangyu Liu, Guy Emerson, and Nigel Collier. Visual spatial reasoning. *Transactions of the Association for Computational Linguistics*, 11:635–651, 2023. doi: 10.1162/tacl_a_00566.

[21] Xinhang Liu, Yu-Wing Tai, and Chi-Keung Tang. ChatCam: Empowering camera control through conversational ai. In *Advances in Neural Information Processing Systems*, volume 37, pages 54483–54506, 2024. doi: 10.52202/079017-1726.

[22] Naila Murray, Luca Marchesotti, and Florent Perronnin. AVA: A large-scale database for aesthetic visual analysis. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2408–2415, 2012. doi: 10.1109/CVPR.2012.6247954.

[23] Tobias Nägeli, Lukas Meier, Alexander Domahidi, Javier Alonso-Mora, and Otmar Hilliges. Real-time planning for automated multi-view drone cinematography. *ACM Transactions on Graphics*, 36(4):1–10, 2017. doi: 10.1145/3072959.3073712.

[24] Shufeng Nan, Mengtian Li, Sixiao Zheng, Yuwei Lu, Han Zhang, and Yanwei Fu. Mind-of-director: Multi-modal agent-driven film previsualization via collaborative decision-making, 2026. arXiv:2603.14790.

[25] Pablo Pueyo, Juan Dendarieta, Eduardo Montijano, Ana C. Murillo, and Mac Schwager. CineMPC: A fully autonomous drone cinematography system incorporating zoom, focus, pose, and scene composition. *IEEE Transactions on Robotics*, 40:1740–1757, 2024. doi: 10.1109/TRO.2024.3353550. arXiv:2401.05272.

[26] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A platform for embodied ai research. In *IEEE/CVF International Conference on Computer Vision*, pages 9339–9347, 2019.

[27] Ilias Stogiannidis, Steven McDonagh, and Sotirios A. Tsaftaris. Mind the gap: Benchmarking spatial reasoning in vision-language models, 2025. arXiv:2503.19707.

[28] Hossein Talebi and Peyman Milanfar. NIMA: Neural image assessment. *IEEE Transactions on Image Processing*, 27(8):3998–4011, 2018. doi: 10.1109/TIP.2018.2831899.

[29] Sheyang Tang, Armin Shafiee Sarvestani, Jialu Xu, Xiaoyu Xu, and Zhou Wang. Aesthetic camera viewpoint suggestion with 3d aesthetic field, 2026. arXiv:2602.20363.

[30] Jiayu Wang, Yifei Ming, Zhenmei Shi, Vibhav Vineet, Xin Wang, Yixuan Li, and Neel Joshi. Is a picture worth a thousand words? delving into spatial reasoning for vision language models, 2024. arXiv:2406.14852; NeurIPS 2024.

[31] Fei Xia, Amir R. Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson Env: Real-world perception for embodied agents. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9068–9079, 2018.

[32] Zhenran Xu, Longyue Wang, Jifang Wang, Zhouyi Li, Senbao Shi, Xue Yang, Yiyu Wang, Baotian Hu, Jun Yu, and Min Zhang. FilmAgent: A multi-agent framework for end-to-end film automation in virtual 3d spaces, 2025. arXiv:2501.12909.

[33] Yuzhe Yang, Liwu Xu, Leida Li, Nan Qie, Yaqian Li, Peng Zhang, and Yandong Guo. Personalized image aesthetics assessment with rich attributes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19861–19869, 2022. doi: 10.1109/CVPR52688.2022.01928.

[34] Gengze Zhou, Yicong Hong, and Qi Wu. NavGPT: Explicit reasoning in vision-and-language navigation with large language models, 2023. arXiv:2305.16986.

A Structured Reviewer Scoring and Feedback

The Reviewer combines deterministic projection checks with VLM-based visual judgment. The internal score used during search contains six signals:

Table 4: **Internal Reviewer signals.** The paper-level notation separates two rule-side geometric signals from four VLM image-side scores.

Paper notation	Code field	Source	Meaning
m_1	`rule_m1`	Blender projection	Whether the primary subject is inside the frame and in the coarse screen half requested by the composition rule.
m_2	`rule_m2`	Blender projection	Normalized distance from the projected subject center to the target composition point.
m_3	`m1`	VLM Reviewer	Composition quality of the rendered preview.
m_4	`m2`	VLM Reviewer	Technical image quality, including readability and rendering artifacts.
m_5	`m3`	VLM Reviewer	Aesthetic quality and photographic appeal.
m_6	`m4`	VLM Reviewer	Alignment with the language instruction and soft blueprint.

For the two rule-side signals, Blender projects the primary subject center into normalized screen coordinates (u, v). If the subject center is outside $[0, 1] \times [0, 1]$, or if a left/right/top/bottom composition preference is violated at the half-screen level, $m_1 = 0$; otherwise $m_1 = 1$. The target point for m_2 is $(0.5, 0.5)$ by default and moves to the corresponding third point for rule-of-thirds preferences. The score is $m_2 = \max(0, 1 - d/0.45)$, where d is the Euclidean screen-space distance from (u, v) to the target point.

For the four VLM signals, the Reviewer receives the candidate camera parameters and the rendered preview image. It must return JSON fields `m1`, `m2`, `m3`, `m4`, `reasoning`, and `summary`. The implementation clamps each score to $[0, 1]$; if parsing fails, the candidate receives a neutral fallback score of 0.5 on all four VLM dimensions. The internal scalar used for search is

$$J(c) = 0.10m_1 + 0.10m_2 + 0.15m_3 + 0.15m_4 + 0.25m_5 + 0.25m_6. \tag{6}$$

This score is used only for internal ranking, region-memory updates, and search diagnostics; all main results in the paper use external post-hoc image metrics.

The Reviewer also produces structured language feedback for the next round. Given all candidate records in a round, it outputs a JSON object with `round_review`, `next_strategy`,

step_scale, explore_ratio_next, preferred_motion, failure_tags, forbidden_zones, and optional seed candidates. The implementation clamps step_scale to $[0.4, 1.8]$, clamps explore_ratio_next to $[0.1, 0.8]$, keeps at most six failure tags, accepts at most two Reviewer-generated forbidden zones, normalizes candidate camera parameters, and merges Reviewer forbidden zones with Reflector dead regions. Pairwise incumbent selection is handled separately: each new preview is compared against the current incumbent, and a parsing failure falls back to keeping the incumbent. These constraints make the Reviewer a schema-bounded search controller rather than an unconstrained conversational critic.

B Model Configuration, Prompts, and Schemas

Table 5 lists the six role-conditioned calls. All share the recorded backbone gpt-5.4, the same agent class, and an OpenAI-compatible endpoint; none uses in-context demonstrations, and all decoding parameters were left at provider defaults. Prompt hashes are SHA-256 prefixes of the exact runtime strings released with the code.

Table 5: **Role configuration.** Each row is one prompt-conditioned call of the same backbone.

Role	Inputs	Outputs	Demos	Hash (8)
Soft-blueprint Director	instruction; reference object; scene object/topology text	primary subject, weighted context objects, composition/camera/target preferences, vibes, negatives	0	3c88f663
Candidate Director	state, blueprint, memory, anchors, context image	candidate position / look-at / lens / aperture / ratio / rationale	0	e5895bce
Single-image Reviewer	one preview image; instruction	four $[0, 1]$ scores, reasoning, summary	0	031b48c8
Pairwise Reviewer	incumbent and challenger images	winner $\in \{A, B\}$, per-dimension winners, reason	0	987a3fd5
Round Reflector	round records; up to four previews	strategy, step scale, explore ratio, failure tags, forbidden zones	0	2f5bed6b
Final-framing Director	winning preview; allowed ratios	aspect ratio and compositional rationale	0	5cb1e791

The operative clauses of the six zero-shot prompts, translated from the released originals, are as follows. *Soft-blueprint Director:* compress the user instruction into a soft blueprint without overhardening the task; identify the primary subject; select at most four preferred context objects with weights; extract composition, camera-angle, camera-zone, look-toward, axis, symmetry, semantic-vibe, and negative preferences; do not invent objects; return JSON only. *Candidate Director:* propose candidate camera moves for the next iteration using the current image and camera, the blueprint, the proposal plan, seed candidates, scout references, and history; respect forbidden and dead regions; produce meaningfully different conservative, moderate, and relocation candidates; keep the subject readable; choose the aspect ratio deliberately and justify it; return JSON only. *Single-image Reviewer:* evaluate one candidate image and do not select a winner or decide next-round control; score composition, technical quality, aesthetics, and instruction alignment in $[0, 1]$; name the main strength and the largest failure; return JSON only. *Pairwise Reviewer:* make a binary choice only, with no averaging, compromise, tie, or third answer; compare composition, technical quality, aesthetics, instruction alignment, subject readability, and instruction compliance; prefer the more stable image that can still be improved; return winner=A|B, per-dimension winners, and a reason. *Round Reflector:* do not score one image or choose a winner; read the whole round, identify its main success and failure patterns, decide between refinement and exploration, and emit the next-round

step scale, exploration ratio, preferred motion, failure tags, up to two forbidden zones, and optional seed candidates. *Final-framing Director:* choose one allowed aspect ratio for the current best image from subject relationships, atmosphere, symmetry, perspective depth, environmental breadth, and visual concentration, with a compositional rationale. Malformed outputs are schema-validated and completed with deterministic seed candidates; a failed parallel preview receives one serial rerender.

C Anchor Bank Construction and Query Stability

The anchor bank is assembled before local search from three families: scene-bounding-box cameras along global axes, sides, and a high overview, with canonical or blueprint targets as look-at points; visibility-grid anchors placed around canonical subject and context objects; and relocation anchors derived from the scout camera poses. Each entry stores position, look-at, lens and aspect hints, source region, and a prior. The entry check verifies only that position, look-at, and focal length exist and are finite; it is not a collision, occlusion, visibility, or semantic-validity test, so "valid" below means structurally well-formed.

To test query stability we repeated the identical production blueprint query five times on each of ten scenes spanning complexity 1–5 (50 queries). All 50 calls returned non-empty banks and 100% of anchors were structurally well-formed under the finite-field check. Raw free-form primary-subject strings fell outside the scene's canonical object vocabulary in 70% of queries, so raw naming is not reliable. After the pipeline's deterministic canonicalization and fallback, pairwise agreement of the primary subject was $.960$, and pairwise Jaccard agreement was $.713$ for preferred context and $.943$ for composition preferences. Under top-10 anchor matching, aggregate differences were $.031$ scene diagonals in position, 6.33° in viewing direction, and $.04$ mm in lens, with outliers reaching $.264$ scene diagonals and 49.19° in two of the ten scenes. Because fallback can map an unmatched phrase to the reference object or the first mesh, these numbers show engineering stability after validation, not semantic correctness: stability is not correctness, and deterministic validation is what makes the bank executable.

D Complexity Rubric and Retrospective Agreement

Table 6: **Benchmark composition.** The same set of 47 scenes summarized along three diversity axes. Each scene contains three tasks, yielding 141 tasks in total.

Category	Subcategory	#Scenes	Total tasks
Visual style	Stylized / Cartoon	16	48
	Realistic	15	45
	Fantasy / Mystical	9	27
	Sci-Fi / Cyberpunk	7	21
Environment	Outdoor / Natural	18	54
	Indoor / Interior	12	36
	Abstract / Mixed	12	36
	Space / Cosmic	5	15
Subject type	Panoramic / No hero	19	57
	Architecture	15	45
	Nature / Object	7	21
	Character / Creature	6	18
Benchmark total		47	141

The complexity label uses four dimensions fixed before benchmark scoring: geometric density (object number, detail, scale, occlusion), material complexity (variety, advanced shaders), lighting complexity (source count, interactions, atmospheric effects), and compositional richness (subject count, depth, relations, viewpoint ambiguity). Level 1 is a minimalist single-object scene without an environment; Level 2 has few objects and basic lighting; Level 3 has moderate complexity and complete environmental context; Level 4 has dense props, advanced materials, multi-source lighting, and richer depth and relations; Level 5 covers urban-scale or film-quality assets with extreme density,

procedural or volumetric effects, and highly viewpoint-dependent readability. The frozen distribution over 47 scenes is $\{1, 10, 14, 16, 6\}$ for levels 1–5 (mean 3.34), and the 90-task evaluation subset spans levels 2–5.

Independent per-rater records from the original annotation stage were not retained, so an original-stage agreement statistic cannot be recovered. We therefore performed a retrospective two-rater re-annotation of all 47 scenes with the frozen rubric and report it explicitly as post-hoc: exact agreement on 43/47 scenes (91.5%), within one level on 47/47, unweighted Cohen's $\kappa = .884$, quadratic-weighted $\kappa = .959$, Spearman $\rho = .964$, and a mean absolute difference of .085 levels. All four disagreements are adjacent-level boundary cases. These statistics support the reproducibility of the aggregate five-level label; they are not original-stage agreement and do not provide dimension-level kappas. Complexity remains descriptive metadata and never enters method ranking.

E Metric-Weight Sensitivity

M_{pq} was rebuilt from the raw IAA, IQA, and ISTA columns only; legacy composite and internal fields were not used. Because the composite is linear in its weights, we enumerated all 5,151 non-negative weight triples on the .01 simplex. Under the paper weights, PhotoFlow and Single-Chain score .5782 and .5674; under equal weights they score .5759 and .5636. PhotoFlow ranks strictly first in 5,085/5,151 settings (98.7%), and the complete ranking coincides with the paper-weight ranking in 4,258/5,151 settings (82.7%). The analytic difference between the two methods is

$$\Delta(\omega) = .0194306\,\omega_a + .0195417\,\omega_q - .0022539\,\omega_s,$$

so the 66 exceptions lie in the ISTA-dominant corner ($\omega_s \geq .90$), not only at a single vertex. The paired bootstrap interval under paper weights is still $[-.0074, .0288]$, and no reweighting turns this image-level composite into a spatial metric.

F Spatial Audit Protocol

The audit covers the 59 valid subject-placement and relational-composition missions that are common-completed across all five methods. One intended spatial mission was excluded because its stored instruction is a duplicate atmosphere prompt with no spatial requirement, and we excluded it rather than inventing a post-hoc target. Three constraints per task were frozen from the instruction text before any image was shown: presence and non-occlusion of the requested primary or environmental subject, the requested screen placement, scale and framing, and the requested viewpoint or camera angle. Two raters independently labeled every image–constraint pair as Yes/No/Unclear with method identities hidden and images presented in randomized order; each rater produced $295 \times 3 = 885$ judgments (1,770 in total). Raw agreement is 70.4%, Cohen's $\kappa = .486$, and nominal Krippendorff's $\alpha = .482$; rater unclear rates are 3.4% and 13.2%. Table 7 reports the per-method breakdown, including the stricter all-three and joint (quality-threshold and spatial) success rates.

Table 7: **Full spatial audit.** Yes= 1, No/Unclear= 0; 59 images per method. "All three" requires every frozen constraint; "Joint" requires $M_{\mathrm{pq}} \geq .55$ and all three.

Rater	Method	Mean sat.	All three	Joint	Any spatial No	Visibility
A	Single-Step	.384	.169	.068	.797	.424
A	Anchor Bank	.215	.068	.034	.915	.254
A	Random	.192	.068	.017	.898	.271
A	Single-Chain	.672	.441	.305	.458	.746
A	PhotoFlow	.661	.508	.339	.458	.712
B	Single-Step	.384	.237	.153	.644	.441
B	Anchor Bank	.271	.136	.068	.746	.237
B	Random	.260	.169	.119	.763	.288
B	Single-Chain	.655	.458	.322	.356	.678
B	PhotoFlow	.605	.441	.288	.441	.610

Correlations between M_{pq} and mean constraint satisfaction, pooled over all methods and tasks ($N = 295$), are Pearson .468 and Spearman .486 for rater A, and Pearson .535 and Spearman .545

for rater B. Because these pooled correlations do not model within-task clustering, we treat the direct violation counts as the stronger evidence: among PhotoFlow images with $M_{\mathrm{pq}} \geq .55$, 38.9% (rater A) and 36.1% (rater B) receive at least one explicit spatial No, versus 59.6% and 45.6% across all methods, and the corresponding rates in the top composite quartile are 60.8% and 48.6% across methods. A high-M_{pq} output can therefore violate key geometric requirements, which is why the paper reports quality and spatial correctness as separate axes.

G Runtime and Cost Accounting

Table 8: **End-to-end cost** on three preselected EEVEE subject-placement scenes (complexity 2/3/5, one run each; Blender 4.4.3, A800 80 GB). Times are measured from immediately before Blender process launch to successful exit with the final image and method log present.

Method	n	Mean / median E2E (s)	Previews	Calls	Tokens	Rounds
Single-Step LLM	3	43.7 / 52.4	1.0	1.0	6,375	$1/1/1$
Single-Chain Reflection	3	199.6 / 203.7	6.0	19.0	82,203	$6/6/6$
PhotoFlow	3	890.9 / 795.7	20.0	47.0	153,800	$6/4/5$

Two of the three PhotoFlow timing runs stopped before round six under the production plateau stop (rounds 6/4/5), so these numbers are as-executed latencies under a maximum-six-round policy, not three complete six-round runs. No significance test is appropriate for $n = 3$. A fixed-batch replay of four first-round candidates with 1/2/4 external Blender workers (three measured repetitions after one discarded warm-up) produced speedups of 0.35–$0.70\times$ rather than gains, because repeated scene loading dominates; we report this negative result and identify persistent render workers, batched rendering, and adaptive candidate budgets as the path to efficiency.

H Failure Taxonomy and Per-Category Results

The 34 PhotoFlow tasks below the quality threshold are the quality-failure subset of the 90 common completed tasks, not an additional filtering step. Two annotators labeled each with one primary and up to two secondary tags from six families. Agreement on a unique primary cause was only 11.8%, so the paper reports non-exclusive tag-presence ranges across raters rather than a single-cause distribution: local search stagnation or redundant regions in 6–7/34 cases, low perceptual quality (dark or noisy) in 14–21, style or atmosphere mismatch in 7–16, subject absent or severely occluded in 6–7, wrong placement/scale/framing in 4–6, and wrong viewpoint or spatial relation in 11–19. For illustration, the forest subject-placement case is a successful escape: both searches start poorly, but PhotoFlow's four-way branching finds an internal-score .656 view in round two and refines it to .743, whereas Single-Chain's six proposals stay at or below .337 despite its own review diagnosing subject loss; both human raters mark PhotoFlow as satisfying subject visibility and the requested trunk/canopy framing and Single-Chain as failing both.

Table 9 asks whether the improvement is dominated by a single mission type. PhotoFlow is first and Single-Chain second in every category: subject placement ($.578$ vs. $.560$), relational composition ($.582$ vs. $.577$), and atmosphere/style ($.574$ vs. $.565$). The margin is largest on subject placement, where relocation and region memory have the most room to act, and smallest on relational composition, where the requested spatial relation itself is the hard part; the lower-order ranking among non-iterative baselines shifts by category.

I Additional Search and Human Evidence

Figure 4 contrasts what the two neighbouring tasks evaluate. VLN scores an instruction-following path over a navigation graph, where the stopping position and the trajectory both matter, whereas virtual photography scores the final rendered camera state and treats the path that produced it as irrelevant. The VLN panel is schematic rather than a run: the thumbnails mark what a navigation policy would optimize (a traversable route and a stop pose) against the view our agent optimizes. Table 9 asks the complementary question of whether the method ordering survives the mission split, and it does for the top two methods.

Table 9: **Per-category results.** External composite M_{pq} on the 90 common completed tasks, split by mission type (30 tasks each). The table tests whether improvements hold across mission types instead of being dominated by a single scene family.

Method	Subject placement	Relational composition	Atmosphere/style
Single-Step LLM	.499	.526	.517
Anchor Bank Best-of-N	.510	.525	.522
Random Search	.514	.523	.544
Single-Chain Reflection	.560	.577	.565
PhotoFlow	**.578**	**.582**	**.574**

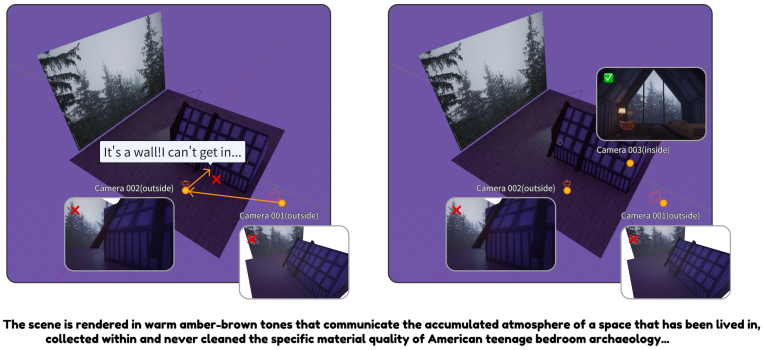

Figure 4: Task boundary with vision-and-language navigation. VLN and virtual photography are neighboring language-conditioned 3D tasks, but VLN evaluates navigation paths and instruction fidelity while virtual photography evaluates the final rendered camera view.

Table 10 gives the full numbers behind the consistency check summarized in Section 5.4. Stage 1 is a forced choice over method pairs and records ties separately (12.67% for aesthetics, 15.33% for alignment), so the selection rates do not sum to one; Stage 2 rates PhotoFlow outputs only and therefore describes absolute quality rather than a preference margin. The 24 rated outputs are the same images scored by the automatic evaluators, which is what makes the MOS correlation meaningful.

Table 10: Human consistency study. Stage 1 reports selection rates; Stage 2 reports PhotoFlow-only MOS and correlation with automatic metrics.

Evaluation	Responses / points	**PhotoFlow**	Strongest non-ours	Additional statistic
Aesthetic selection	450	**31.33%**	24.89% single-chain	ties 12.67%
Alignment selection	450	**30.22%**	23.11% single-chain	ties 15.33%
PhotoFlow MOS	300	**3.208** mean	–	aesthetic 3.313, alignment 3.103
M_{pq} vs. MOS mean	24	Pearson .827	Spearman .697	PhotoFlow-only ratings

Three readings of Table 10 are worth making explicit. The selection rates are not a ranking of image quality: a vote for PhotoFlow means it was preferred over a specific opponent for that mission, so the percentages describe pairwise wins against the strongest non-ours method rather than absolute preference. Ties are reported separately because a forced choice would otherwise inflate both rates, and at 12.67% and 15.33% ties the effective margins are smaller than the raw percentages suggest. The MOS column rates PhotoFlow outputs only, so it measures the absolute plausibility of our renders rather than a margin over a competitor; its agreement with the automatic composite (Pearson .827, Spearman .697 over 24 images) is the evidence that the composite tracks human judgement, and the small sample is why we treat it as a consistency check rather than a population-level validation.

Figure 5 shows the case behind the high-explore ablation. Without the lane all sixteen previews stay in the dim region above the ground plane, so the incumbent barely improves; the Reflector marks those cells as unproductive and the Director is pushed to an anchor on the other side of the subject, which

produces the brighter and better-separated final render on the right. This is the escape behaviour counted by the coverage and collapse diagnostics, not a source of uniformly better proposals.

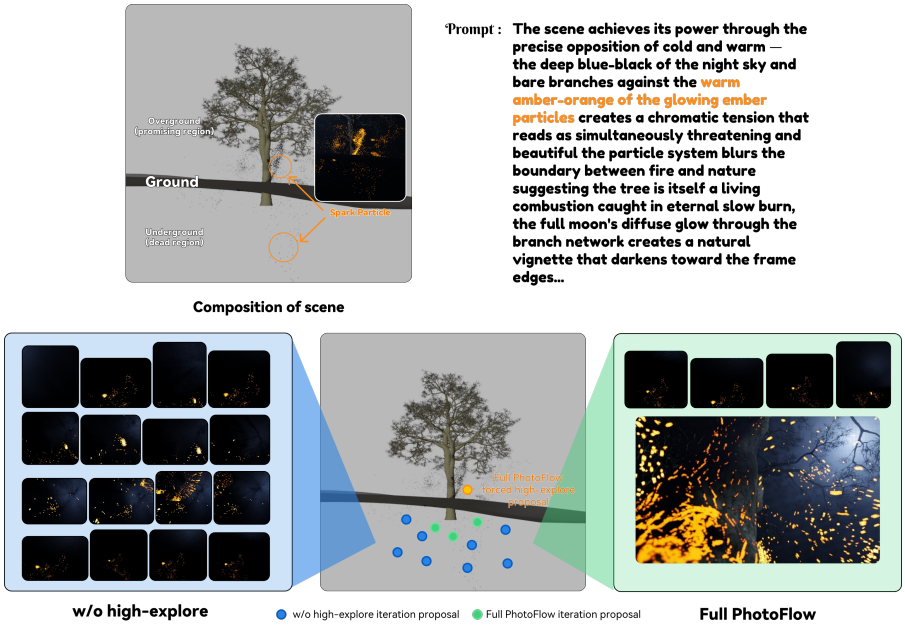

Figure 5: High-explore as a switchable safeguard. A representative case where forced high-explore helps the search leave a locally acceptable but weak viewpoint and find a stronger composition. This component is intended as an escape route from local collapse, not as a universally better proposal source.

J Qualitative Case Studies

Figures 6 and 7 show qualitative cases in the same layout: the left column gives the language prompt, the middle column shows the iterative search previews, and the right column shows the final selected render.

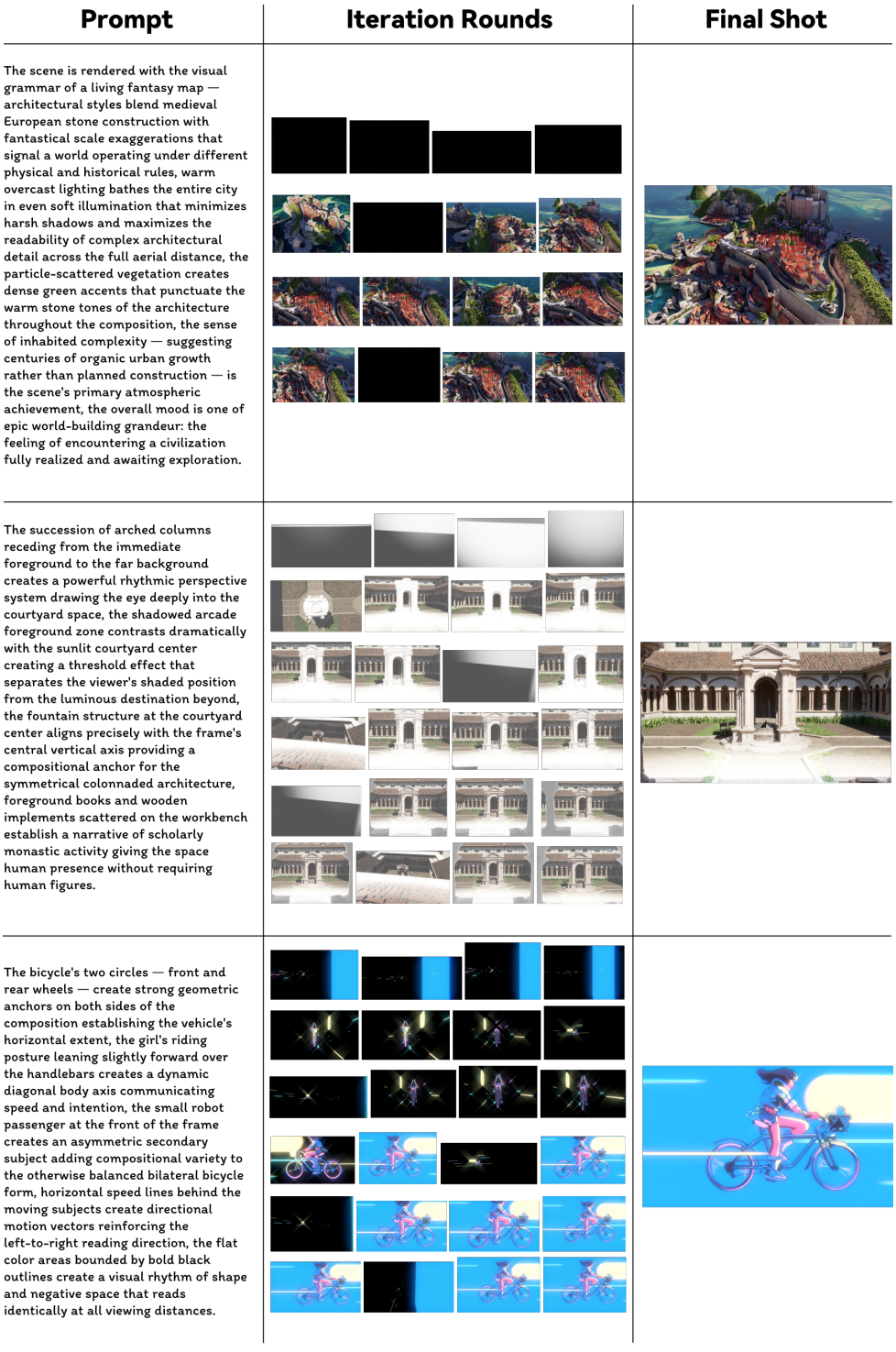

Figure 6: Successful qualitative cases. Each row is organized as prompt, iterative previews, and final render. The three examples cover a city/island composition, a courtyard architecture view, and a stylized bicycle subject, showing how PhotoFlow turns language into a sequence of rendered camera hypotheses and a final executable camera state across different scene scales and visual styles.

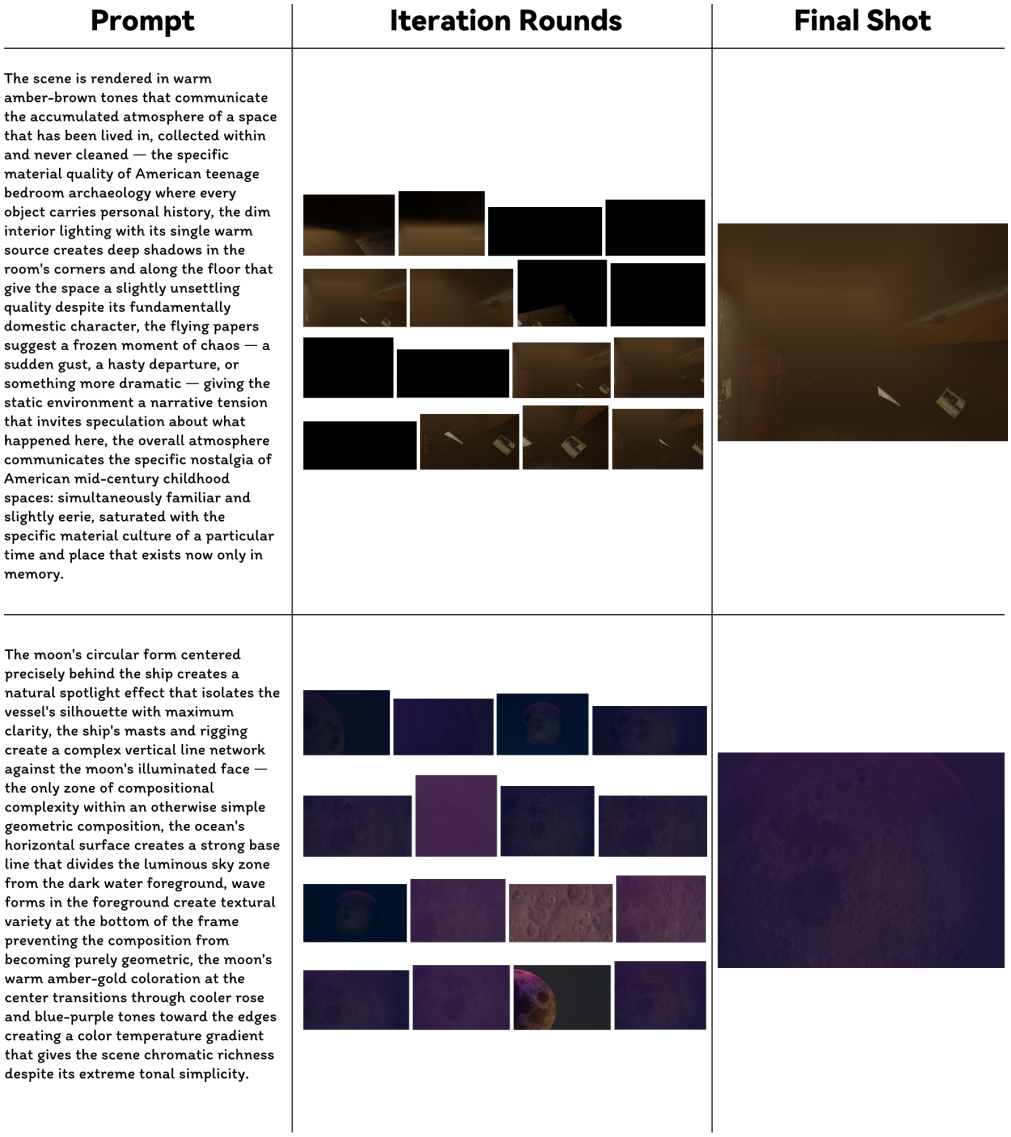

Figure 7: Failure qualitative cases. Each row is organized as prompt, iterative previews, and final render. The top row is `037_attic_hideout_atmosphere_style`: the search collapses into a dark, low-quality atmospheric view and receives a hard-failure tag with constraint satisfaction 0.0 ($M_{\text{pq}} = .244$). The bottom row is `031_medieval_ship_ocean_scene_subject_placement`: the final camera fails the requested subject/framing constraint despite partial semantic alignment, also yielding constraint satisfaction 0.0 ($M_{\text{pq}} = .338$).

Both rows come from the hard-failure log rather than being selected for effect: the failure tag, the constraint satisfaction and M_{pq} are copied from the same record as the render, so the classification can be checked against the image. The top row collapses into a dark, low-contrast view; the bottom row keeps the scene semantics but misses the requested subject and framing. In both cases the composite and the constraint check agree, which is the easy case. The disagreement that motivates the spatial audit in Section 5.3 is the opposite one, where a high-M_{pq} render still violates the frozen geometry.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Section 1 state the benchmark size, the agent design, the held-out protocol, the modest and statistically inconclusive gain over the strongest iterative baseline, and the separate spatial audit. All quantitative claims are tied to Tables 1–3 and the appendix tables, and the internal Reviewer score is explicitly treated as a process diagnostic rather than final evidence.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 6 discusses anchor-bank dependence, the internal Reviewer as a noisy search proxy, the image-level composite's blind spot for geometric correctness, the modest and uncertain aggregate gain with its $4.46\times$ wall-clock cost, and the gap between controlled virtual photography and physical or dynamic camera systems.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper contains no theorems or proofs. The equations define the mission tuple, the executable camera state, the perceptual-quality composite, the internal search score, and the high-explore priority.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Section 5.1 specifies the 24 development and 117 held-out missions, the 90-task common-completed set and its 27 method-independent exclusions, the search budgets and as-executed rounds, and the baselines; Section 3.7 and Appendix B give the recorded backbone, provider-default decoding, the six zero-shot prompts, schemas, and prompt hashes. Appendix A documents the internal scoring, Appendix E the weight-simplex enumeration, Appendix F the spatial audit protocol, and Appendix G the cost accounting.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Agent code, benchmark registry, task specifications, per-scene source and license metadata, complete prompts, and evaluation scripts are released at `https://github.com/Visionary-Laboratory/PhotoFlow`.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 5.1 gives the held-out and development splits, the common-completed filtering rule, the budget and as-executed rounds, the compared methods, and the external evaluators; Section 3.7 and Appendix G list hardware, render settings, preview and final sample counts, model calls, and tokens.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Section 5.2 reports paired sign tests, McNemar tests, and paired bootstrap intervals for the PhotoFlow/Single-Chain comparison, and states explicitly that the aggregate difference is not statistically conclusive; Appendix E reports the exhaustive weight-simplex enumeration.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Section 3.7 and Appendix G report Blender 4.4.3 on one A800 80 GB GPU, preview and final sample counts, model calls, tokens, and mean/median end-to-end wall-clock times for three methods on three scenes.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The study uses rendered virtual scenes, automatic image metrics, and anonymous preference and constraint judgments over rendered images. No deployment for physical robots, surveillance, or safety-critical systems is claimed.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The work targets creative camera-control tools in virtual environments. Section 6 discusses the limits of the claims, and Section 4 documents the heterogeneous per-asset licenses of the scene assets and the risk of over-reading image-level perceptual metrics.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The benchmark does not introduce safety-critical deployment, biometric identification, medical, legal, or other high-risk decision making.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All 47 scenes are Creative-Commons-licensed Blender assets with heterogeneous per-asset terms. The released registry records per-scene source URL, author, license, and verification date, Appendix D lists the scene composition, and original assets remain governed by their upstream licenses.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The benchmark contributes language-conditioned task specifications, structured evaluation constraints, region-memory and log schemas, and rendered outputs derived from open-license scenes; Section 4 and Appendix D describe the task schema, mission types, and composition.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

Justification: Section 5.4 and Appendix I report a two-stage study in which 30 of 31 participants, after quality control, contributed 450 pairwise preference responses and 300 Likert ratings; Section 5.3 and Appendix F add 1,770 anonymous image–constraint judgments from two raters.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [No]

Justification: No formal IRB approval was obtained. Both protocols collected only anonymous preference and constraint judgments over rendered images: participants provided no personal or sensitive information, no risks beyond ordinary screen viewing were involved, and there was no interaction beyond rating.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: LLMs and VLMs are core components: one recorded multimodal backbone serves as soft-blueprint Director, candidate Director, single-image Reviewer, pairwise Reviewer, round Reflector, and final-framing Director. Section 3.7 and Appendix B document the prompts, output schemas, prompt hashes, zero-shot setting, and fallback logic.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.