Check for updates

Designing in Context: A Scene-Contextual 2D-to-3D Generative AI Pipeline for Conversational XR Space Design

Seonghoon Ban*
RECON Labs Inc., Seoul, Korea

Abstract

Background Conversational interfaces and generative artificial intelligence(AI) are opening extended reality (XR) space design to creators without 3D expertise: a user describes an object aloud and sees it appear in the scene. In current practice, each request is served by a text-to-3D model that generates the object in isolation, blind to the scene it will join. The result is a familiar failure: objects that clash with the scene's style, arrive at implausible scales, and ignore the lighting. Graphics research shows that routing generation through 2D image priors yields visually coherent scenes, but such pipelines run offline, outside any authoring loop. Missing is a generation strategy that is both scene-coherent and conversationally interactive, and evidence that the difference matters.

Methods We propose scene-contextual generation, a methodology in which each requested object is created inside the scene rather than apart from it. The system captures the creator's current view, inpaints the object into that image so it inherits the scene's style, lighting, and scale, segments the inpainted object, lifts it to a 3D Gaussian-splat asset, and places it back at a depth-anchored position. We implemented this pipeline and a context-free baseline in one voice-driven authoring system built on Gaussian-splat scenes, and conducted a within-subjects study in which design-trained participants (N = 15) furnished two stylistically contrasting environments (a low-poly fantasy campground and a photoscanned living room) under each generation strategy (2 × 2, order-balanced). Measures combined in-app ratings of coherence, satisfaction, agency, and workload after each condition, comparative forced choices, behavioral logs of latencies and corrective actions, and think-aloud walkthrough interviews.

Results Scene-contextual generation was rated more coherent (M = 5.36 vs. 4.18; $F(1,14) = 5.84$, $p = .030$, $\eta p^2 = .29$), more satisfying, and less effortful, with no significant strategy × scene interaction. The advantage held equally in both environments, and direct generation was weakest in the photoscanned room. Fourteen of fifteen participants chose the contextual strategy for continued use in the campground and thirteen in the living room (binomial $p = .001$, $.007$), despite its longer latency (46 vs. 29 s per request).

Conclusions Scene-contextual generation reframes what a generative XR design tool should produce: not an object, but an object-in-relation-to-a-scene. The findings position 2D image priors as a practical harmonization layer for conversational 3D creation and yield design implications for generative spatial-design tools.

Keywords Generative AI, Extended Reality (XR), Spatial Design, 3D Object Generation, Visual Coherence

This work was supported by the Ministry of Culture, Sports and Tourism and the Korea Creative Content Agency through the Culture Technology Research and Development Program (RS-2024-00393915).

*Corresponding author: Seonghoon Ban (vincent@reconlabs.ai)

Citation: Ban, S. (2026). Designing in Context: A Scene-Contextual 2D-to-3D Generative AI Pipeline for Conversational XR Space Design. *Archives of Design Research, 39*(4), 53-69.

http://dx.doi.org/10.15187/adr.2026.11.39.4.53

Received : Aug. 12. 2026 ; **Reviewed :** Aug. 24. 2026 ; **Accepted :** Aug. 24. 2026
pISSN 1226-8046 **eISSN** 2288-2987

Copyright : This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/3.0/), which permits unrestricted educational and non-commercial use, provided the original work is properly cited.

1. Introduction

Extended reality (XR) is becoming a medium in which spaces are not only visited but made. Yet authoring a coherent 3D environment still demands modeling skill, asset literacy, and spatial-composition judgment that most people — and many designers — do not have. A growing body of work therefore couples generative AI with natural-language interfaces so that a creator can simply say what should exist: large language models assemble interactive scenes from prompts (De La Torre et al., 2024), speech-driven systems generate 3D content inside headsets (Weng et al., 2024; Behravan & Gračanin, 2025; Lee et al., 2025), and elicitation studies show that people spontaneously address such agents conversationally, pointing at the world around them (Aghel Manesh et al., 2024). Voice suits XR authoring particularly well: hands remain free for spatial manipulation, and the creator never has to leave the scene being designed.

The prevailing architecture behind these systems is what we call context-blind generation: each spoken request is passed to a text-to-3D model or an asset retriever, and the result is dropped into the scene. The object is created in isolation, with no knowledge of the space it will join. The visual consequences are familiar to anyone who has used such a tool: a photorealistic chair lands in a low-poly fantasy campground; a lamp arrives at dollhouse scale; materials ignore the room's light. These are not incidental artifacts. Systems that compose independently sourced assets — retrieved (Yang et al., 2024) or procedurally parameterized (C. Sun et al., 2025; Hu et al., 2024) — have no mechanism by which an object's appearance can be conditioned on the scene's, and a recent comparison of speech-based generation pipelines found directly generated objects judged lowest in perceptual quality (Xiu et al., 2025). The design cost compounds: every added object erodes the scene's stylistic unity, and the burden of harmonizing falls back on precisely the person the tool promised to relieve.

Computer graphics research points to an alternative. Image generation models are, in effect, harmonization engines: when an object is inpainted into an image, it inherits the surrounding style, lighting, perspective, and implied scale from local pixel context, at no extra cost. Recent systems exploit this by routing 3D scene generation through 2D imagery — expanding worlds tile by tile through contextual inpainting (Engstler et al., 2025), reconstructing object-decomposed scenes from a single photograph (Yao et al., 2025; Huang et al., 2025), and populating environments by inpainting objects into scene renders before lifting them to 3D (Wang et al., 2024). These pipelines demonstrate that a 2D prior can carry coherence into 3D. But they run offline, as batch processes that deliver a scene; no creator converses with them, and none has been designed or evaluated as an interactive authoring loop.

This paper closes that gap. We propose scene-contextual generation: a methodology for conversational XR space design in which each requested object is generated inside the creator's current view of the scene rather than apart from it. When the creator asks for an object, the system captures the current viewpoint, inpaints the object into that image — where it inherits the scene's look — segments the inpainted region, lifts it to a 3D Gaussian-

splat asset, and places it back at a depth-anchored position and scale. We implemented this pipeline alongside a context-free direct path in a single voice-driven system, so the two strategies can be compared under an identical interface. Figure 1 previews the contrast: the same campground corner furnished under each strategy.

We evaluate the methodology where it is meant to live: inside the authoring loop. In a within-subjects study, fifteen design-trained graduate students furnished two stylistically contrasting environments — a low-poly fantasy campground and a photoscanned, Gaussian-splat living room — under both generation strategies (2 × 2, order-balanced), rating coherence, satisfaction, agency, and workload after each condition and comparing the strategies directly at the end, while the system logged every utterance, generation latency, and corrective action. Crossing generation strategy with scene style also lets us ask not only whether context inheritance matters, but where it matters most.

This paper contributes:

- Scene-contextual generation — a methodology that routes each object request through view capture, in-context inpainting, segmentation, image-to-3D lifting, and depth-anchored placement, so new objects inherit the scene's style, lighting, and scale;
- a working voice-conversational XR authoring prototype implementing both this pipeline and direct text-to-3D generation under one dialogue interface;
- empirical evidence from a within-subjects authoring study (N = 15) on whether — and where — context inheritance improves perceived coherence and creator preference, joined with behavioral traces, and distilled into design implications for generative spatial-design tools.



Figure 1 The same campground corner furnished under (a) direct and (b) scene–contextual generation

2. Related Work

2. 1. Language–Driven 3D Scene Construction

A first family of systems treats a scene as an arrangement of independently sourced parts, with language models deciding what goes where. 3D-GPT dispatches cooperating LLM agents to drive procedural generators (C. Sun et al., 2025); SceneCraft has an LLM write Blender code and critique renders of its own output in a self-improvement loop (Hu et al., 2024); Holodeck has GPT-4 propose spatial constraints solved over assets retrieved from Objaverse (Yang et al., 2024); LayoutGPT emits numeric layouts directly (Feng et al., 2023). Later systems refine placement with vision-language feedback (LayoutVLM; F.-Y. Sun et al., 2025) or unify multiple synthesis tools under a self-reflective agent (SceneWeaver; Yang et al., 2025). Whatever the sophistication of the arrangement, the parts remain visually independent: retrieval bounds appearance to what a library contains, and generation-from-text knows nothing of the scene's palette, materials, or light. Where coherence is addressed at all, it is structural — layout and adjacency — not appearance-level. Our work targets the appearance level.

2. 2. Image–First 3D Generation

A second family reaches coherence by generating imagery first. Text2Room grows a room mesh through iterative inpainting and depth estimation (Höllein et al., 2023). SynCity extends a world tile by tile, generating each new tile's image within the rendered context of its neighbors before lifting it to 3D (Engstler et al., 2025). CAST decomposes a single photograph into per-object 3D reconstructions aligned by pose and physical constraints (Yao et al., 2025), and MIDI generates all objects of a scene jointly from one image (Huang et al., 2025). Closest to our method, Architect populates environments by inpainting objects into scene renders, perceiving them with vision models, and lifting placements via monocular depth (Wang et al., 2024). Object-level editing systems apply the same logic to existing 3D scenes: GaussianEditor inpaints an object into one rendered view, reconstructs it with image-to-3D, and inserts it into a Gaussian-splatting scene in roughly five minutes per operation (Chen et al., 2024); InseRF performs text-driven insertion into neural fields (Shahbazi et al., 2024); FreeInsert, conversely, argues for insertion without 2D spatial priors (Li et al., 2025).

Collectively, these systems establish the premise our method borrows — 2D priors harmonize. None, however, operates as a conversational, in-situ authoring loop. Architect is an offline population pipeline whose principal furniture is retrieved rather than generated, so the inpainted appearance is only approximately honored — a diversity limit its authors acknowledge (Wang et al., 2024). GaussianEditor is a desktop editing operation, not an authoring dialogue. And none has been evaluated with human judges of scene coherence.

2. 3. Generative Authoring in XR and Its Evaluation

Authoring from inside the medium is a long-standing ambition, from CavePainting (Keefe et al., 2001) and immersive tangible-AR authoring (Lee et al., 2004) to consumer tools of the Tilt Brush era. Generative AI reframed it. LLMR builds and modifies interactive mixed-reality scenes from prompts in real time, though its content comes from code and existing

assets (De La Torre et al., 2024); DreamCodeVR lets speech edit object behavior (Giunchi et al., 2024); VRCopilot studies human–AI co-creation of room layouts in VR, showing that intermediate representations shape user agency (Zhang et al., 2024). In-headset object generation has arrived — Dream Mesh (Weng et al., 2024), Matrix (Behravan & Gračanin, 2025), and ImaginateAR, which pairs text-to-image with image-to-3D generation in outdoor AR (Lee et al., 2025) — but each employs a single pipeline and evaluates usability rather than coherence.

Two recent studies motivate ours directly. Say It, See It compared four speech-driven generation pipelines in AR (n = 11, three fixed prompts) and found text-to-image-to-3D routes superior to direct text-to-3D on perceived quality (Xiu et al., 2025) — yet its objects were judged in isolation, with no scene to cohere with. CoGen3D found that creators value an intermediate concept image as a checkpoint in conversational 3D asset creation (Jiang et al., 2026). What remains untested is the combination that defines real spatial authoring: generation conditioned on the scene and judged in the scene. Table 1 locates this gap.

Table 1 Related systems by scene conditioning and authoring loop

	Context–blind generation	Scene–contextual genera–tion
Offline / batch	3D–GPT; SceneCraft; Ho–lodeck; LayoutGPT; Lay–outVLM; SceneWeaver	Text2Room; SynCity; CAST; Architect; MIDI; GaussianEditor; InseRF
Interactive /in–situ	LLMR; Dream Mesh; Ma–trix; ImaginateAR; VRCo–pilot (layout); Say It, See It	This work

3. Scene–Contextual Generation Pipeline

3. 1. Design Rationale

Three principles, distilled from the failures reviewed above, drive the design.

DP1 — Context inheritance. The appearance, scale, and lighting of a new object should be inherited from the scene itself, not specified through parameters or prompts. Asking creators to verbalize a scene's style ('watercolor, muted palette, soft afternoon light…') re-imposes exactly the expertise the tool should absorb; the pixels already carry that information.

DP2 — Conversational locality. In immersive authoring, people request objects deictically, relative to where they stand and look (Aghel Manesh et al., 2024). The creator's current view is therefore not merely an image source but the design context: the composition within which the new object must work.

DP3 — Graceful latency. A multi-stage generative pipeline cannot be instantaneous. Rather than hiding latency, the system should remain conversational across it — acknowledging the request, showing a provisional proxy, and materializing the final asset asynchronously — so that waiting becomes part of the authoring rhythm rather than a rupture in it.

3. 2. Pipeline

Figure 2 summarizes the pipeline as seven stages.

S1 — Utterance understanding. Speech is transcribed and parsed by an LLM into an object specification and a placement intent. This stage is shared verbatim with the baseline (Section 3.4), anchoring the comparison.

S2 — View capture. The system captures the creator's current view as an RGB image together with the engine's metric depth buffer. Because metric depth comes from the renderer, later placement is anchored in true scene scale rather than in estimated scale alone — a practical advantage over fully estimate-based pipelines such as Architect's (Wang et al., 2024).

S3 — Contextual inpainting. A placement region within the captured view is proposed by a vision-language model from the parsed intent, and the requested object is inpainted into that region; an automatic verification pass accepts the edit or retries it. This stage realizes DP1: surrounding pixels condition the generation, so the object arrives already carrying the scene's style, lighting, perspective, and an implied size.

S4 — Object segmentation. The inpainted object is isolated with promptable segmentation (SAM 3; Carion et al., 2026), prompted with the object description and its proposed placement region, yielding a clean object crop and its 2D footprint.

S5 — 3D lifting. The segmented cutout is first re-rendered by the image model as a clean, studio-lit single-object view — preserving the designed appearance while normalizing background and exposure for reconstruction robustness — and then lifted to a 3D Gaussian-splat asset. Because the scene itself is a Gaussian-splat environment, generated objects share the scene's representation, avoiding the visual seam of inserting textured meshes into splats. Routing through an image intermediate is independently supported (Xiu et al., 2025).

S6 — Scale and pose resolution. Dense monocular depth of the edited view is affine-aligned to the engine's metric depth; the asset's world position and scale are then solved from the aligned depth at the anchor and the footprint of the inpainted mask — the size the image model chose within the composition is treated as the intended size. The object is snapped to its supporting surface by raycast.

S7 — Placement and refinement loop. The moment placement is decided, a holographic proxy with a live progress label appears at the target location and is swapped for the finished asset when ready (DP3); up to three requests proceed in parallel, and the dialogue remains open for refinement ('bigger', 'move it left', 'try again').

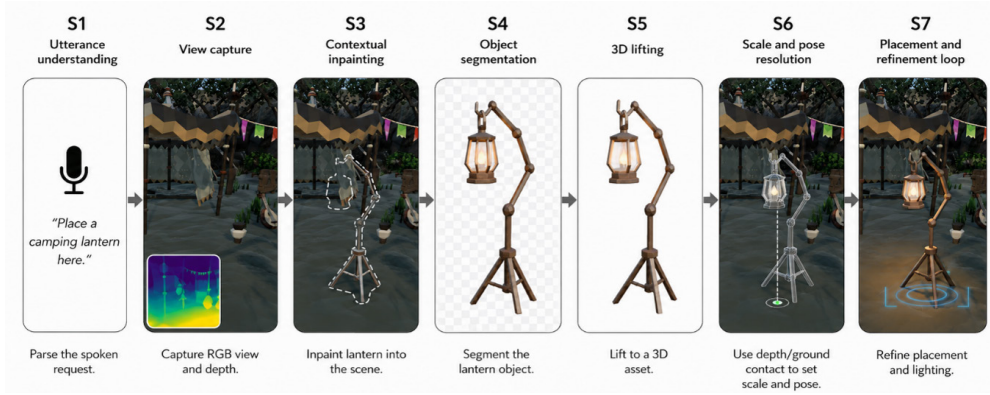

Figure 2 The seven–stage scene–contextual pipeline, shown with actual intermediate outputs

3. 3. Implementation

The system is implemented in Unity (URP) on Gaussian-splat scenes. Spoken interaction — recognition and conversational feedback in Korean — uses the GPT Realtime API (gpt-realtime-2.1; OpenAI, 2025); placement decisions and edit verification use Gemini 3.5 Flash-Lite (Google, 2026a), accessed via OpenRouter; contextual inpainting and object-image enhancement use Nano Banana 2 Lite, a lightweight model in Google's Gemini image family (Google, 2026b); segmentation uses SAM 3 (Carion et al., 2026); and 3D lifting uses TripoSplat (VAST AI Research, 2026), which outputs Gaussian-splat assets. All generative inference runs over cloud APIs; up to three requests proceed in parallel, and a per-request cost ceiling guards against runaway spending.

When a stage fails by predefined criteria (an empty segmentation mask; a degenerate reconstruction), the pipeline reports the failure conversationally and re-attempts; failure rates are reported with the study. Stage-level latency, aggregated from the study logs, appears in Table 2.

Table 2 Per–stage latency from study logs (M (SD), seconds; 255 requests)

Stage	M (SD), s	Notes
S1–S2 Capture and placement decision	0.1 + 2.9 (1.6)	direct: 2.8 (1.1); shared reasoning stage
S3 Contextual inpainting + verification	8.5 (1.9) + 2.4 (0.6)	contextual only
S4 Segmentation	6.6 (9.2)	contextual only
S5 Enhancement + 3D lifting	8.4 (2.2) + 16.9 (5.0)	direct: text–to–image 6.9 (1.8) + 3D 19.4 (8.4)
S6–S7 Scale, pose, and placement	0.7 (0.4)	direct: 0.1 (0.1)
Total — scene–contextual mode	≈ 46	sum of serial stages
Total — direct mode	≈ 29	

3. 4. Baseline: Direct Generation

The baseline represents current practice — context-blind generation. The identical voice interface and S1 parsing route the object specification to a context-free image generation: the object is rendered from the spoken description alone, with no scene information, then lifted by the same TripoSplat stage and placed by the same anchoring, with scale set from estimated real-world dimensions. The two conditions therefore share the image intermediate, the 3D

generator, and the placement machinery; they differ only in whether the object image is conditioned on the scene. This also controls for the known advantage of image-intermediate pipelines over direct text-to-3D (Xiu et al., 2025), isolating context inheritance itself. One asymmetry remains by design: in the contextual pipeline, scale emerges from the inpainted composition (S6), whereas the baseline estimates it symbolically. This is a constitutive property of each strategy, not a confound to be removed; we state it explicitly and return to it in the results.

4. User Study

4. 1. Design

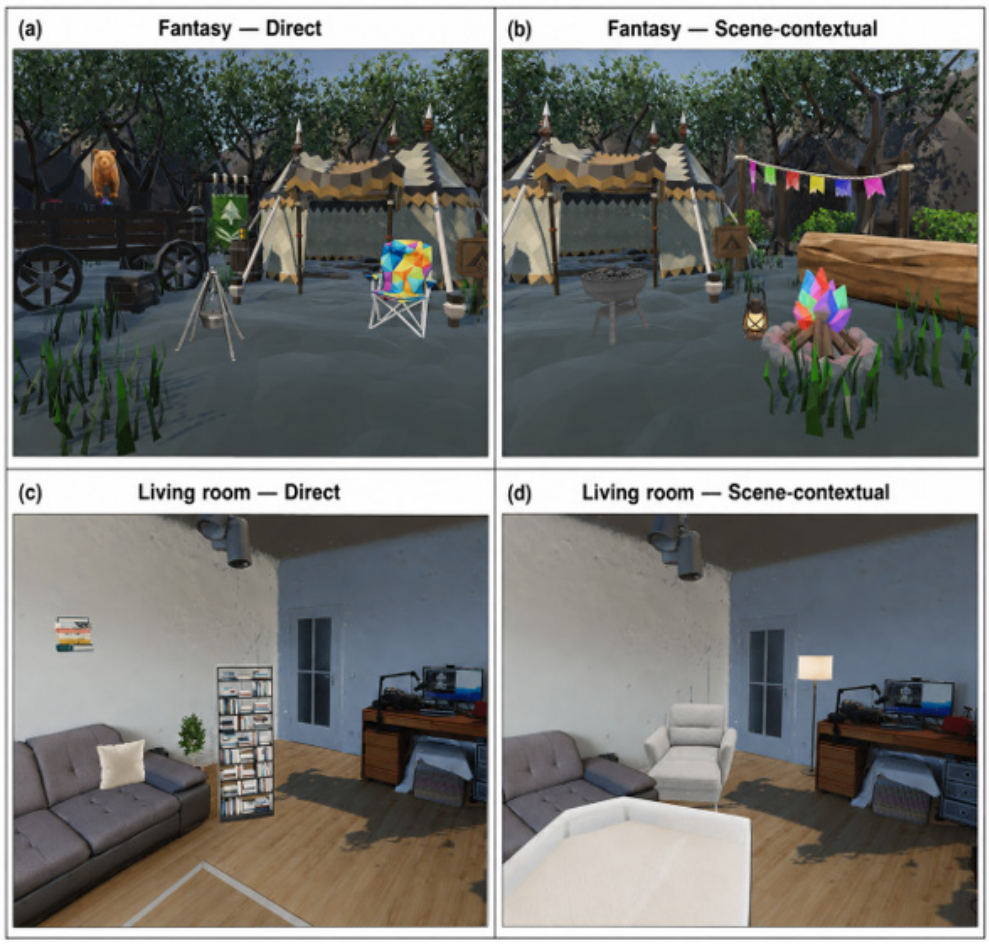

Figure 3 One participant's four corners: two scenes crossed with two strategies



Figure 4 Representative generated objects across participants, by scene and strategy

The study is a 2 × 2 within-subjects experiment crossing generation strategy (direct vs. scene-contextual) with scene style: a low-poly fantasy campground, where the stylistic distance from generic model output is large, and a photoscanned living room reconstructed as Gaussian splats, where it is small. Each participant furnished one corner of each environment under each strategy — four cells in total (Figure 3) — with cell order assigned from a Williams 4 × 4 balanced Latin square. Owing to a participant-ID assignment error at scheduling, the realized order distribution deviated from the planned three-per-order balance (ABCD: 2; BDAC: 6; CADB: 4; DCBA: 3); an order-sensitivity check is reported in Section 5.1. Because a creator necessarily knows which strategy is active, the design makes no blinding claim: it

measures the experienced outcome, and triangulates self-ratings with forced comparative choices and behavioral traces (Section 4.4). Condition labels were nonetheless hidden from the participant-facing interface.

Apparatus: a desktop standalone build of the system, navigated in first person with mouse and keyboard and spoken to via push-to-talk in Korean.

4. 2. Participants

Fifteen design-trained participants — graduate students in design-related programs (e.g., spatial, interaction, or media design) — were recruited from outside the author's organization to avoid conflicts of interest. None had used the system before. Before each session, participants were informed of the study's purpose and procedures, of the data to be collected (questionnaire responses, interaction logs, and audio recordings), and of their use for research purposes; participation was voluntary and without compensation, and all participants gave informed consent. No sensitive personal data were collected. As the author's institution is a private company without an institutional ethics board, the study was not subject to formal IRB review; the procedures above were followed to uphold standard ethical safeguards for human-subjects research.

4. 3. Task and Procedure

After a short training block in a practice area (excluded from analysis), each cell asked the participant to furnish the assigned corner by voice: three fixed requests — drawn from a per-scene pool of six object categories, split between the two strategies and swapped across participants so that object–strategy pairings were balanced — plus one free request, within an eight-minute limit or until the participant declared the corner finished. Generation proceeded live; participants could refine, retry, or nudge results by voice and keys, and all such actions were logged. Representative objects generated across participants are shown in Figure 4. After each cell, participants answered a 13-item in-app questionnaire (Table 3), followed by two brief on-screen prompts answered aloud (the most and the least satisfying aspect of the corner), captured through the system's speech transcription. After all four cells, a comparative block asked for the best and the most disappointing corner, a forced choice of strategy for continued use in each scene with a confidence rating, and two further spoken prompts (choice rationale; perceived strategy difference); the session closed with a think-aloud walkthrough of the four corners and a semi-structured interview (about 15 minutes). A session lasted 75–90 minutes.

4. 4. Measures

Per-cell questionnaire (7-point Likert, administered in Korean; English glosses in Table 3): perceived coherence (six items; internal consistency reported as Cronbach's α), outcome satisfaction (two), sense of agency (two), workload and frustration (two, reverse-scored), and latency acceptability (one). Comparative block: best and most disappointing corner, per-scene strategy choice with confidence. Behavioral logs, captured automatically by the system: per-stage generation latency, failures and retries, spawn counts, manual corrections (position, scale, and rotation nudges), and time per cell. All utterances — including responses to the spoken prompts — and the closing walkthrough were recorded for qualitative analysis, and every pipeline artifact (captures, inpainted frames, cutouts, assets) was archived per request.

Table 3 Per-cell in-app questionnaire (7-point Likert; administered in Korean, English glosses shown)

Subscale	Item
Coherence (α reported)	The generated objects match the overall style of this space
Coherence	The objects look as if they had originally belonged in this space
Coherence	Their colors and materials fit the surroundings
Coherence	Their sizes look natural
Coherence	Their lighting and brightness fit the space
Coherence	Overall, the objects are in harmony with the space
Satisfaction	I am satisfied with this finished corner
Satisfaction	I would like to show this result to others
Agency	I felt in control of the creation process
Agency	The result reflects my intent
Workload (reversed)	Creating this corner was mentally demanding
Workload (reversed)	The process felt frustrating
Latency	The waiting time for generation was acceptable

4. 5. Analysis

Subscale composites enter 2 (strategy) × 2 (scene style) repeated-measures analyses — ANOVA, corroborated by Wilcoxon signed-rank tests where normality fails (cf. Wobbrock et al., 2011) — with coherence designated the primary outcome, Holm correction applied across the secondary subscales, and effect sizes emphasized given the sample size. Per-scene strategy choices are tested against chance with exact binomial tests. Behavioral measures (latency, retries, corrections, time) are compared with paired tests and reported descriptively. Interviews, spoken-prompt responses, and think-aloud recordings are analyzed through reflexive thematic analysis (Braun & Clarke, 2006, 2019), conducted by the author; consistent with the reflexive approach, no inter-coder reliability is computed, and themes are reported with supporting quotations.

5. Results

5. 1. Perceived Coherence

The six-item coherence composite was highly consistent (Cronbach's $\alpha = .96$). Scene-contextual generation was rated more coherent than direct generation ($M = 5.36$, $SD = 1.19$ vs. $M = 4.18$, $SD = 1.58$; $F(1,14) = 5.84$, $p = .030$, $\eta p^2 = .29$; Wilcoxon $p = .041$), and the campground was rated more coherent than the photoscanned room overall ($F(1,14) = 7.99$, $p = .013$, $\eta p^2 = .36$). The strategy × scene interaction was not statistically significant ($F(1,14) = 0.01$, $p = .94$): the contextual advantage was virtually identical in the stylized campground ($\Delta = +1.16$) and in the photoscanned room ($\Delta = +1.20$), where direct generation received its lowest ratings ($M = 3.69$, $SD = 1.53$). Individual variability was substantial — ten of fifteen participants rated the contextual corner higher in the campground and eight in the living room (SD of the paired difference ≈ 2.1) — which the qualitative accounts in Section 5.4 help interpret. Strategy effects did not differ across order groups (Kruskal–Wallis $p = .46$). — Figure 5a, Table 4.

5. 2. Preference, Satisfaction, and Agency

Comparative choices were near-unanimous: for continued use, 14 of 15 participants chose the contextual strategy in the campground and 13 of 15 in the living room (exact binomial $p = .001$ and .007; choice confidence $M = 6.40$, $SD = 1.06$). The best-rated corner was most often the contextual campground (9/15) and the most disappointing the direct living room (10/15). Satisfaction ($F(1,14) = 8.74$, $p = .010$, $\eta p^2 = .38$) and sense of agency ($F(1,14) = 6.04$, $p = .028$, $\eta p^2 = .30$) likewise favored the contextual strategy; after Holm correction among the secondary subscales, satisfaction and workload remained significant and agency was marginal (adjusted $p = .055$). Notably, the reverse-scored workload composite also favored the contextual strategy (6.47 vs. 6.00; $F(1,14) = 9.18$, $p = .009$, $\eta p^2 = .40$): creating with it was experienced as less effortful, not more. — Figure 5b, Table 4.

5. 3. Behavior and Latency

Mean per-request latency was 46 s under the contextual strategy and 29 s under the direct strategy (Table 2, Figure 5c), yet the 17-second premium did not depress rated latency acceptability (5.97 vs. 5.43; $p = .052$, in the contextual strategy's favor). Participants issued 255 generation requests (130 contextual, 125 direct), spawned comparable numbers of objects per cell ($M = 4.8$ vs. 4.7), and curated actively in both strategies (63 vs. 62 deletions overall). Manual nudging was used by about half of the participants (8 and 7 of 15), with somewhat more adjustment episodes per cell under the contextual strategy ($M = 10.5$ vs. 7.1). Mean authoring time per cell was 258 s vs. 175 s, well under the eight-minute cap; pipeline retries were rare (3), and 13 stage failures were all recovered conversationally.

5. 4. Qualitative Findings

Reflexive analysis of the spoken responses produced three themes. (T1) Coherence is read as absorption. Contextually generated objects were praised for melting into the scene — 'the size and style of everything matched the space so well' (A04); several participants articulated the manipulation unprompted: 'with direct, it felt like an image had been taken as-is and turned into 3D; in the other corners the textures blended into the screen' (A03), 'direct did not consider the context at all' (A07). (T2) Realism raises the bar. The photoscanned room punished small deviations that the stylized campground forgave: 'in the realistic room, even a slight oddity felt wrong — the campground was more forgiving' (A13); participants noticed that 'existing objects carry shadows and lighting, but the generated ones didn't (A08) — an account that helps explain why direct generation was weakest there. (T3) Placement, not appearance, is the residual friction. Across both strategies, complaints concentrated on orientation, attachment, and overlap ('the chandelier ended up inside the wall', A14) and on the difficulty of expressing spatial intent verbally: 'I couldn't put that distance into words' (A11); 'I wished I could move what had already been created' (A09). Appearance-level dissatisfaction, by contrast, was voiced almost exclusively about the direct strategy.

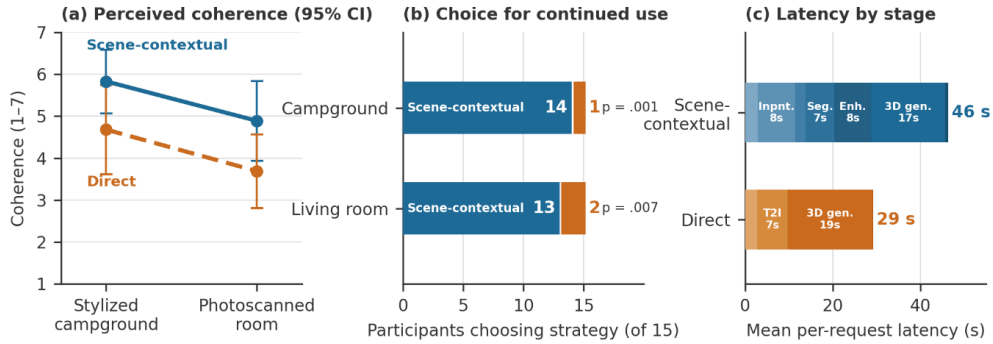

Figure 5 (a) Coherence interaction; (b) per–scene strategy choice; (c) per–request latency by stage

Table 4 Subscale means by strategy, pooled over scenes (N = 15; df = 1, 14)

Subscale	Contextual M (SD)	Direct M (SD)	Strategy effect
Coherence (α = .96)	5.36 (1.19)	4.18 (1.58)	F = 5.84, p = .030, ηp^2 = .29
Satisfaction	5.13 (1.40)	3.77 (1.49)	F = 8.74, p = .010, ηp^2 = .38
Agency	5.28 (1.22)	4.28 (1.59)	F = 6.04, p = .028, ηp^2 = .30
Workload (reverse–scored)	6.47 (0.88)	6.00 (0.94)	F = 9.18, p = .009, ηp^2 = .40
Latency acceptability	5.97 (1.13)	5.43 (1.43)	F = 4.51, p = .052, ηp^2 = .24

6. Discussion

6. 1. Design Implications

Generate relations, not objects. The unit of generation in spatial design tools should be the object-in-scene, not the object. Where current toolchains treat the scene as a container into which independently generated assets are dropped, our findings show that conditioning generation on the scene's own pixels changes what creators judge their space to be — unified rather than assembled. Context is not metadata attached to a request; it is material from which the object is made.

The 2D prior as a harmonization layer. The mechanism is architectural — capture, inpaint, segment, lift, place — and agnostic to which inpainting or image-to-3D model fills each slot. As underlying models churn, the pattern survives: any sufficiently capable image model can serve as the harmonization layer for 3D creation. This reframes how design-tool builders can absorb generative-model progress without redesigning the authoring experience.

Context inheritance is a general principle, not a stylized-scene fix. We expected the contextual advantage to concentrate where the scene departs most from generic model output; instead it was equally large in the photoscanned room — where direct generation was weakest — and the interaction was not statistically significant. Generic objects clash with any coherent environment, photoreal ones included. Tool builders should treat scene conditioning as a default of generative spatial authoring rather than an optional mode for stylized worlds. And although the contextual strategy took longer — a 17-second premium per request —

participants did not perceive creating with it as more demanding: the workload measure in fact favored it.

6. 2. Limitations

Four limitations bound our claims. First, creators rated their own outputs, and the active strategy cannot be blinded in live use — a property inherent to experience studies of authoring tools (cf. Zhang et al., 2024); the forced comparative choices, the behavioral traces, and the archived scenes — which permit a later blind third-party rating — temper but do not remove this. Second, the apparatus was a desktop first-person build; the judged attributes are visible under free navigation, but immersive scale and presence differ, and headset replication is warranted before strong XR-specific claims. Third, breadth: two scenes, seventeen requests per participant on average, and one implementation instance; object–strategy pairings were balanced, but coverage of object categories and scene genres is necessarily partial. Fourth, the panel is fifteen design-trained graduate students rather than senior practitioners, the realized condition orders were imbalanced (Section 4.1) — though strategy effects did not vary by order — and qualitative coding was performed by a single researcher under a reflexive approach; effect sizes are emphasized over p-values throughout.

6. 3. Future Work

Four directions follow. A blind third-party rating of the archived participant scenes would complement self-assessment with independent judgment, and semantic style metrics computed over the archived generation artifacts — if they track coherence judgments — could let a metric-in-the-loop variant self-assess and regenerate before presenting results. The placement frictions of Section 5.4 point to hybrid refinement: voice for creation, direct manipulation for layout. Because the pipeline conditions on a single captured view, coherence is inherited from the creator's viewpoint at the moment of request, yet the generated object will later be seen from other positions and angles; how far single-view inheritance holds across viewpoints — and how future systems might condition on multi-view or scene-level 3D context — is a natural next question. And the capture–inpaint–lift pattern invites extension beyond single-object insertion, toward multi-object edits and scene-wide restyling within the conversation; a headset deployment study incorporating creativity-support instruments (Cherry & Latulipe, 2014) remains the natural step for the full XR claim.

7. Conclusion

Conversational spatial authoring will be judged by whether the spaces it produces feel like spaces rather than collections. We proposed scene-contextual generation — capture, inpaint, segment, lift, place — as a way for 2D image priors to carry a scene's character into every newly created 3D object; implemented it alongside a context-free baseline in one voice-driven system; and compared the two strategies in a within-subjects authoring study with design-trained participants. Creators rated contextually generated objects as more coherent in both a stylized and a photoscanned environment, chose the contextual strategy almost unanimously for continued use, and found creating with it less — not more — effortful

despite its longer latency. Beyond the specific models employed, the contribution is a design pattern for generative spatial design: the scene is not the destination of a generated object but the condition of its generation.

Acknowledgement
This work was supported by the Ministry of Culture, Sports and Tourism and the Korea Creative Content Agency through the Culture Technology Research and Development Program (RS-2024-00393915).

References

1. Aghel Manesh, S., Zhang, T., Onishi, Y., Hara, K., Bateman, S., Li, J., & Tang, A. (2024). How people prompt generative AI to create interactive VR scenes. In *Proceedings of the 2024 ACM Designing Interactive Systems Conference (DIS '24)* (pp. 2319–2340). ACM. https://doi.org/10.1145/3643834.3661547
2. Behravan, M., & Gračanin, D. (2025). From voices to worlds: Developing an AI-powered framework for 3D object generation in augmented reality. In *2025 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)* (pp. 150–155). IEEE. https://doi.org/10.1109/VRW66409.2025.00038
3. Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology, 3*(2), 77–101. https://doi.org/10.1191/1478088706qp063oa
4. Braun, V., & Clarke, V. (2019). Reflecting on reflexive thematic analysis. *Qualitative Research in Sport, Exercise and Health, 11*(4), 589–597. https://doi.org/10.1080/2159676X.2019.1628806
5. Carion, N., Gustafson, L., Hu, Y.-T., Debnath, S., Hu, R., Suris, D., Ryali, C., Alwala, K. V., Khedr, H., Huang, A., Lei, J., Ma, T., Guo, B., Kalla, A., Marks, M., Greer, J., Wang, M., Sun, P., Rädle, R., … Feichtenhofer, C. (2026). *SAM 3: Segment anything with concepts. In International Conference on Learning Representations (ICLR 2026)*. https://proceedings.iclr.cc/paper_files/paper/2026/hash/e0982cbc81401df3430ee1ff780dc7a2-Abstract-Conference.html
6. Chen, Y., Chen, Z., Zhang, C., Wang, F., Yang, X., Wang, Y., Cai, Z., Yang, L., Liu, H., & Lin, G. (2024). GaussianEditor: Swift and controllable 3D editing with Gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2024)* (pp. 21476–21485). IEEE. https://doi.org/10.1109/CVPR52733.2024.02029
7. Cherry, E., & Latulipe, C. (2014). Quantifying the creativity support of digital tools through the Creativity Support Index. *ACM Transactions on Computer-Human Interaction, 2*1(4), Article 21. https://doi.org/10.1145/2617588
8. De La Torre, F., Fang, C. M., Huang, H., Banburski-Fahey, A., Amores Fernandez, J., & Lanier, J. (2024). LLMR: Real-time prompting of interactive worlds using large language models. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems (CHI '24)* (Article 600, pp. 1–22). ACM. https://doi.org/10.1145/3613904.3642579
9. Engstler, P., Shtedritski, A., Laina, I., Rupprecht, C., & Vedaldi, A. (2025). SynCity: Training-free generation of 3D worlds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV 2025)* (pp. 27585–27595). IEEE. https://doi.org/10.1109/ICCV51701.2025.02561
10. Feng, W., Zhu, W., Fu, T.-J., Jampani, V., Akula, A., He, X., Basu, S., Wang, X. E., & Wang, W. Y. (2023). LayoutGPT: Compositional visual planning and generation with large language models. In *Advances in Neural Information Processing Systems, 36, 18225–18250 (NeurIPS 2023)*. https://doi.org/10.52202/075280-0802
11. Giunchi, D., Numan, N., Gatti, E., & Steed, A. (2024). DreamCodeVR: Towards democratizing behavior design in virtual reality with speech-driven programming. In *2024 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)* (pp. 579–589). IEEE. https://doi.org/10.1109/VR58804.2024.00078
12. Google. (2026a). *Gemini 3.5 Flash-Lite* [Large language model]. Google DeepMind. https://deepmind.google/models/gemini/flash-lite/

13. Google. (2026b). *Nano Banana 2 Lite* (Gemini image model) [Image generation and editing model]. Google DeepMind. https://blog.google/innovation-and-ai/models-and-research/gemini-models/gemini-omni-flash-nano-banana-2-lite/
14. Höllein, L., Cao, A., Owens, A., Johnson, J., & Nießner, M. (2023). Text2Room: Extracting textured 3D meshes from 2D text-to-image models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV 2023)* (pp. 7875–7886). IEEE. https://doi.org/10.1109/ICCV51070.2023.00727
15. Hu, Z., Iscen, A., Jain, A., Kipf, T., Yue, Y., Ross, D. A., Schmid, C., & Fathi, A. (2024). SceneCraft: An LLM agent for synthesizing 3D scenes as Blender code. In *Proceedings of the 41st International Conference on Machine Learning (ICML 2024)*, (Vol. 235, pp. 19252–19282). PMLR. https://dl.acm.org/doi/abs/10.5555/3692070.3692846
16. Huang, Z., Guo, Y.-C., An, X., Yang, Y., Li, Y., Zou, Z.-X., Liang, D., Liu, X., Cao, Y.-P., & Sheng, L. (2025). MIDI: Multi-instance diffusion for single image to 3D scene generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2025)* (pp. 23646–23657). IEEE. https://doi.org/10.1109/CVPR52734.2025.02202
17. Jiang, W., He, W., Tan, Z., Kuang, Z., Yu, D., Hasegawa, S., Mayer, S., & Sarsenbayeva, Z. (2026). *CoGen3D: An agentic human-AI co-design pipeline for 3D asset generation for virtual reality.* arXiv. https://doi.org/10.48550/arXiv.2607.03731
18. Keefe, D. F., Feliz, D. A., Moscovich, T., Laidlaw, D. H., & LaViola, J. J., Jr. (2001). CavePainting: A fully immersive 3D artistic medium and interactive experience. In *Proceedings of the 2001 Symposium on Interactive 3D Graphics (I3D '01)* (pp. 85–93). ACM. https://doi.org/10.1145/364338.364370
19. Lee, G. A., Nelles, C., Billinghurst, M., & Kim, G. J. (2004). Immersive authoring of tangible augmented reality applications. In *Third IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2004)* (pp. 172–181). IEEE. https://doi.org/10.1109/ISMAR.2004.34
20. Lee, J., Aleotti, F., Mazala, D., Garcia-Hernando, G., Vicente, S., Johnston, O. J., Kraus-Liang, I., Powierza, J., Shin, D., Froehlich, J. E., Brostow, G., & Van Brummelen, J. (2025). ImaginateAR: AI-assisted in-situ authoring in augmented reality. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25)* (pp. 1–21). ACM. https://doi.org/10.1145/3746059.3747635
21. Li, C., Wang, W., Li, Q., Sebe, N., Lepri, B., & Nie, W. (2025). FreeInsert: Disentangled text-guided object insertion in 3D Gaussian scene without spatial priors. In *Proceedings of the 33rd ACM International Conference on Multimedia (MM '25)* (pp. 10915–10924). ACM. https://doi.org/10.1145/3746027.3755072
22. OpenAI. (2025). *Realtime API* [Software service]. https://platform.openai.com/docs/guides/realtime
23. Shahbazi, M., Claessens, L., Niemeyer, M., Collins, E., Tonioni, A., Van Gool, L., & Tombari, F. (2024). *InseRF: Text-driven generative object insertion in neural 3D scenes*. arXiv. https://doi.org/10.48550/arXiv.2401.05335
24. Sun, C., Han, J., Deng, W., Wang, X., Qin, Z., & Gould, S. (2025). 3D-GPT: Procedural 3D modeling with large language models. In *2025 International Conference on 3D Vision (3DV)* (pp. 1253–1263). IEEE. https://doi.org/10.1109/3DV66043.2025.00119
25. Sun, F.-Y., Liu, W., Gu, S., Lim, D., Bhat, G., Tombari, F., Li, M., Haber, N., & Wu, J. (2025). LayoutVLM: Differentiable optimization of 3D layout via vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2025)* (pp. 29469–29478). IEEE. https://doi.org/10.1109/CVPR52734.2025.02744
26. VAST AI Research. (2026). *TripoSplat: Generative 3D Gaussians with learned density control* [Computer software]. https://www.tripo3d.ai/research/triposplat
27. Wang, Y., Qiu, X., Liu, J., Chen, Z., Cai, J., Wang, Y., Wang, T.-H., Xian, Z., & Gan, C. (2024). Architect: Generating vivid and interactive 3D scenes with hierarchical 2D inpainting. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*. (pp.67575–67603) https://doi.org/10.52202/079017-2158

28. Weng, S. C.-C., Chiou, Y.-M., & Do, E. Y.-L. (2024). Dream Mesh: A speech-to-3D model generative pipeline in mixed reality. In *2024 IEEE International Conference on Artificial Intelligence and eXtended and Virtual Reality (AIxVR)* (pp. 345–349). IEEE. https://doi.org/10.1109/AIxVR59861.2024.00059
29. Wobbrock, J. O., Findlater, L., Gergle, D., & Higgins, J. J. (2011). The aligned rank transform for nonparametric factorial analyses using only ANOVA procedures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)* (pp. 143–146). ACM. https://doi.org/10.1145/1978942.1978963
30. Xiu, Y., Chilukuri, J., Sen, S., & Gorlatova, M. (2025). Say it, see it: A systematic evaluation on speech-based 3D content generation methods in augmented reality. In *2025 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)* (pp. 320–326). IEEE. https://doi.org/10.1109/ISMAR-Adjunct68609.2025.00069
31. Yang, Y., Sun, F.-Y., Weihs, L., VanderBilt, E., Herrasti, A., Han, W., Wu, J., Haber, N., Krishna, R., Liu, L., Callison-Burch, C., Yatskar, M., Kembhavi, A., & Clark, C. (2024). Holodeck: Language guided generation of 3D embodied AI environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2024)* (pp. 16277–16287). IEEE. https://doi.org/10.1109/CVPR52733.2024.01536
32. Yang, Y., Jia, B., Zhang, S., & Huang, S. (2026). SceneWeaver: All-in-one 3D scene synthesis with an extensible and self-reflective agent. In *Advances in Neural Information Processing Systems 38*, 140319–140351 https://doi.org/10.52202/085713-4689
33. Yao, K., Zhang, L., Yan, X., Zeng, Y., Zhang, Q., Xu, L., Yang, W., Gu, J., & Yu, J. (2025). CAST: Component-aligned 3D scene reconstruction from an RGB image. *ACM Transactions on Graphics, 44*(4), Article 86. https://doi.org/10.1145/3730841
34. Zhang, L., Pan, J., Gettig, J., Oney, S., & Guo, A. (2024). VRCopilot: Authoring 3D layouts with generative AI models in VR. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology (UIST '24)* (pp. 1–13). ACM. https://doi.org/10.1145/3654777.3676451