

ANTHONY ROWE, Carnegie Mellon University, USA and Bosch Research, USA



what was said to what was visible and where it occurred. This structured representation enables *SPIN* to identify missing information and gaps in the authored instructions and generate targeted follow-up questions that elicit omitted details, resulting in more complete spatial instructions. In a study with ten participants, *SPIN* significantly outperformed baseline approaches in retrieval performance (mean score: 2.35/3.0), while active elicitation significantly improved graph completeness ($\Delta = 0.68$). Participants reported usability and workload scores of 86.5 on the System Usability Scale and 28.0 on the NASA Task Load Index, respectively.

CCS Concepts: • **Human-centered computing** → Ubiquitous and mobile computing systems and tools; **Mixed / augmented reality**; • **Computing methodologies** → *Spatial and physical reasoning*.

Additional Key Words and Phrases: Spatial Understanding, Knowledge Graphs, Egocentric Video, Active Elicitation, Instruction Authoring, Smart Glasses

ACM Reference Format:

Sruti Srinidhi, Tao Jin, David Lindlbauer, and Anthony Rowe. 2018. *SPIN: Spatial Instruction Authoring from Egocentric Multimodal Data*. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (CHI '27)*. ACM, New York, NY, USA, 25 pages. <https://doi.org/XXXXXXX.XXXXXXX>

1 Introduction

People spend much of their lives navigating physical spaces and performing tasks within them: cooking in a kitchen, running equipment in a lab, repairing machinery in a factory. But communicating how to do these tasks to someone new to the space is surprisingly hard, because the knowledge is inseparable from the space itself: which drawer, which button, which corner of the room, in what order, under what conditions.

Consider a host preparing a home for a short-term guest, like an Airbnb setting. They need to explain where things are across several rooms, the quirks of specific appliances, the multi-step sequence for a finicky hot tub, and situated routines like when to take out the trash or the specific trick to make the water dispenser work.

Today, people handle this with written manuals, static diagrams, or recorded video walkthroughs. However, they are difficult to author, hard to search through, and rigid by nature, meaning they capture only what the author thought to include, with no awareness of the space itself, and often miss details that are actually important.

This creates three persistent problems:

- (1) **High authoring burden:** manually documenting every step and spatial detail in text or video is slow and tedious.
- (2) **Lack of spatial grounding:** since text and images lack spatial context, they expect readers to map the abstract steps onto the 3D space.
- (3) **Instructional omissions:** because authors already understand the process, they may overlook steps or explanations which are actually valuable for another person new to the space.

We address these challenges with *SPIN*, a spatial authoring system that captures instructions as an author walks through a space and narrates it naturally, as though guiding another person new to the space. At the core of *SPIN* is a *Spatially-Aware Knowledge Graph*: rather than storing instructions as video or a flat transcript, it encodes what was said, tied to what was visible at that moment, and where in the space it occurred, linking related information both spatially and semantically. This structure makes the underlying knowledge easy to parse and query, enabling efficient retrieval by location or object rather than scrubbing through linear footage. To populate this graph, *SPIN* captures multimodal egocentric data (synchronized speech and egocentric video) from smart glasses, applying visual re-localization to recover a 3D trace of where the author walked, looked, and spoke. Together, these signals allow the

system to capture not just what was said, but what was pointed at, demonstrated, and spatially referenced, directly addressing the lack of grounding and high authoring burden in existing approaches.

The graph representation also helps identify missing information. Authors giving a walkthrough do not always provide complete instructions: they may omit steps that feel obvious, or skip details they assume are already known. To address this, *SPIN* performs active elicitation, using gaps in the graph to generate targeted follow-up questions that capture what was missed. This closes information gaps at authoring time rather than leaving them to be discovered by a confused user later. Together, these components directly address all three challenges: the graph structure reduces authoring burden and provides spatial grounding, while active elicitation closes the instructional omissions that walkthroughs alone cannot prevent.

Because the resulting knowledge graph is structured and spatially grounded, it is well suited for a range of downstream applications. For example, it can be queried to answer questions about the space, replayed as spatial overlays through an AR headset, or exported into a traditional manual. In this paper, we focus on the authoring side of this pipeline, the process of capturing, structuring, and building more complete spatial knowledge. While our contribution focuses on authoring, we evaluate the resulting graph through conversational querying, using it as a proxy to demonstrate that the captured instructions are more complete and usable.

To inform our system design, we conducted a formative study examining how people naturally give walkthroughs of physical spaces: what information they communicate, the balance of visual, spoken, and deictic (pointing or gesture-based) cues they rely on, and what kinds of details they tend to omit. These findings directly informed the graph schema and active elicitation logic of *SPIN*.

We evaluate *SPIN* through a user study in which 10 participants authored spatial instructions for real physical spaces using our system. We assess the completeness of the resulting knowledge graph before and after active elicitation, the accuracy of instruction retrieval against a benchmark of 20 questions across 5 categories, and the usability of the authoring experience. *SPIN* significantly outperformed video and transcript-based baselines on retrieval accuracy (mean 2.35 vs. 1.53 and 1.20 out of 3.0), while active elicitation produced significant gains in graph coverage and completeness ($\Delta = 0.68$), with authors rating the system 86.5 on the System Usability Scale and 28.0 on NASA-TLX, indicating the system was both easy to use and not cognitively demanding.

Overall, our key contributions are:

- *SPIN*, a spatial authoring system that captures multimodal egocentric data during a natural walkthrough to ground spoken instructions in their physical context.²
- The *Spatially-Aware Knowledge Graph*, a structured representation that links verbal instructions to visual context and physical location to support efficient retrieval.
- An active elicitation approach that identifies and fills gaps in authored instructions using rule-based and LLM-based reasoning, improving completeness without additional unguided effort from the author.
- Results of an evaluation showing that *SPIN* outperforms existing baselines, and that active elicitation significantly improves graph completeness beyond walkthrough alone.

²Code will be made open-source.

2 Related Work

2.1 Traditional vs Situated Instructions

Traditional instructional media, such as printed manuals or 2D video tutorials, fundamentally decouple task information from the physical context in which it must be applied. This separation imposes a significant “mapping problem” on the user [34], who must perform the heavy lifting of translating generic, 2D representations onto their specific 3D environment [45, 58]. This cognitive burden is often attributed to the *split-attention effect*, where the user’s focus is divided between the instructional source and the physical task, leading to increased error rates and frustration [3, 12].

Research in spatial instructions and Augmented Reality tries to address this by anchoring the instructions directly onto the task environment [10, 36, 41, 64]. Early work demonstrated that registering instructions to the user’s field of view can reduce task error rates and improve task localization [16, 32, 61]. Recent efforts have expanded this by moving conventional tutorials, like instruction manuals and videos, into 3D AR guides, allowing users to view demonstrations from any perspective while maintaining spatial registration to physical objects [7, 22, 37, 38].

Despite these benefits, traditional AR assistance remains notoriously brittle due to an authoring bottleneck [40, 63]. These systems often rely on pre-authored, fixed content, making them difficult to scale across the infinite variety of everyday physical tasks [5]. While tools like *ProcessAR* [8] and *InstructAR* [24] have simplified the creation of in-situ guides, they still largely produce static or linear content. Our work builds on these foundations by moving beyond pre-authored sequences, using a generative approach to create dynamic, queryable instructions that are deeply integrated with the user’s immediate physical state.

2.2 Multimodal Scene Understanding & Knowledge Graphs

Recent breakthroughs in Vision-Language Models (VLMs) and open-vocabulary foundation models have fundamentally transformed how systems interpret physical spaces. Foundation models like SAM [46] provides a robust base for object identification and segmentation, while multimodal encoders such as CLIP, and DINO [6, 43] allow for the extraction of high-dimensional semantic features directly from visual data. By leveraging the reasoning capabilities of native multimodal LLMs and VLMs [1, 11, 31], systems can now translate these low-level signals into high-level functional descriptions. These advancements enable a shift from simple label classification to a nuanced understanding of object properties and utility within a broader instructional context.

However, traditional spatial semantic research often relies on geometric representation such as point clouds, meshes, NeRF etc. that are frequently static and require exhaustive, offline pre-scanning [35, 51]. These native 3D pipelines often lack high-order detail and suffer from significant latency, especially when trying to use them for semantic tasks, making them unsuitable for interactive applications requiring immediate adaptation to novel environments [2, 25, 48, 59]. Consequently, leveraging VLMs to extract semantic meaning from renderings has emerged as a more viable approach for constructing **3D-aware representations** that prioritize real-time performance. By anchoring VLM-derived insights into a unified spatial framework, research has demonstrated the feasibility of building language-aligned maps that are searchable and highly detailed without the overhead of traditional 3D reconstruction workflows [14, 18, 42, 52].

To organize these insights for practical utility, researchers increasingly utilize **Knowledge Graphs** and **Semantic Scene Graphs** as primary data structures [21, 23]. Unlike standard 3D maps, these graphs define functional hierarchies and situational relationships between entities, serving as a persistent and queryable record of a walkthrough [15, 27, 50]. Our work aligns with this shift toward *online* scene understanding, where a spatially-aware knowledge graph is constructed incrementally in real-time. Our system builds an active, retrievable structure that facilitates proactive

elicitation and real-time intervention, ensuring the resulting knowledge base is both comprehensive and fully situated in the physical environment.

2.3 Mixed-Initiative & Collaborative Authoring

Instruction by Demonstration has emerged as a more natural method for capturing expert knowledge, allowing authors to narrate tasks while performing them in-situ [4, 8, 26, 33]. By moving authoring from desktop-based environments to embodied performance, these tools significantly lower the technical threshold for content creation [40, 53–55, 63]. However, a critical limitation persists: most current systems act as passive recorders. They rely entirely on the expert to provide a complete and coherent narrative, which is not always the case, experts have a tendency to overlook granular, "intuitive" steps that are actually important for novice comprehension [17]. Consequently, demonstrations captured through passive observation often result in linear sequences with significant information gaps that render the instructions incomplete for complex or variable tasks [38?].

To address these gaps, our work moves toward Mixed-Initiative authoring models, treating the instruction walkthrough as a collaborative dialogue between the human expert and a proactive agent [20, 30]. While early agentic systems were often restricted to rigid, pre-defined task models, modern AI based approaches have introduced the potential for more flexible scene understanding [29, 56, 57]. However, even these recent frameworks primarily focus on generating guidance from a fixed set of user queries rather than proactively shaping the content during the initial demonstration phase. By integrating the principles of Contextual Inquiry[62], our system performs real-time information elicitation, identifying structural or semantic ambiguities and intervening with targeted follow-up questions while the author is still situated in the relevant context [28]. This ensures that the final knowledge base is comprehensive and handles the possible omissions during a human-led walkthrough.

3 Formative Study: Understanding Spatial Information Communication

We conducted a formative study to investigate how spatial instructions are naturally shared and to identify where the communications typically have gaps. Our study was driven by three primary research questions: (a) what types of information do experts prioritize when providing instructions about a space? (b) what information do they frequently omit? and (c) how is this information communication? Is it verbal, visual, deictic, embodied or something else?

3.1 Experimental Design and Data Capture

We recruited four pairs of participants (N=8; 5 male, 3 female adults between 18-65 years of age) to interact in two different spaces: a research lab and a shared kitchen. Each pair consisted of an **expert** who is very familiar and actively used the space and a **novice** who had never been there before. The experts were tasked with providing a walkthrough that would enable the novice to use the space independently, without needing further assistance.

To understand the potential gaps in the instructions provided by the expert, we introduce two different types of tasks:

- **Planned Tasks:** Experts were informed that the novice would later need to do a given task, specifically connect a laptop to a projector (lab) or brew a pot of coffee (kitchen).
- **Unplanned Tasks:** Novices were also privately assigned a second task that the expert was unaware of, which was setting up a workstation (lab) or preparing iced sparkling water (kitchen).

These tasks were designed to test whether experts provide a sufficiently dense body of information to cover more general instructions, or if they provide only narrow, task-specific instructions.

All interactions were recorded using smart glasses, capturing egocentric audio and video from both the expert and the novice. Novices were able to ask follow-up questions during the walkthrough, which we analyzed to identify the specific information gaps that naturally occur.

After the walkthrough, the novices attempted to complete both the planned and unplanned tasks alone. By observing where the novices struggled or succeeded, we identified the specific failure points in manual instructions.

Two of the authors independently coded all recordings to identify recurring patterns, which were then compared and led to the findings explained below. These findings provided the requirements for our system’s active elicitation logic and the structural design of our knowledge graph.

3.2 Formative Study Findings and System Design Principles

Across all four expert-novice pairs, we observed four consistent patterns in how spatial instructions were communicated and where they broke down (present in 4 of 4 sessions for each finding). We use each of these findings to determine design principles we use to build our system.

Finding F1: *Deictic referencing and spatial ambiguity*. Experts relied heavily on physical gestures alongside speech to anchor instructions, often using vague terms like "here" or "this" that were only resolvable by the physical gesture accompanying them (e.g., P4: "Just line it up with these two arrows"). Speech alone was consistently insufficient to convey the instruction, motivating a system that binds transcript to real-time visual and spatial context.

Requirement R1: *Multimodal Grounding*: The system must simultaneously capture textual narration for semantic meaning alongside visual, deictic, and spatial data to resolve location and ensure instructions are physically situated.

Finding F2: *Narrative clutter and task-irrelevant tangents*. Experts frequently digressed into personally interesting but task-irrelevant stories, such as P3’s detour into drone safety history ("some people came up with a clever method to use a fishing wire to tie them to one of the vents"), unrelated to the novice’s immediate needs. This created uneven "layers of importance" within the narration, motivating a system that can distinguish task-critical content from tangential detail.

Requirement R2: *Goal-Directed Information Hierarchy*: The system must maintain an objective, global view of the overarching task to ensure all necessary steps are covered while distinguishing between essential task-critical information and auxiliary details.

Finding F3: *Proximity-based navigation and natural interaction pauses*. Experts navigated spaces by moving to whatever was physically nearest rather than following a fixed task order, and novices consistently used transition points or pauses as their natural window to ask about bypassed objects (e.g., P3 Novice: "What are those things on the wall over there?"). This suggests transition boundaries are the least disruptive point for a system to intervene.

Requirement R3: *Proximity-Aware Interventions*: The system should process a segment only after the author begins to move away from a specific spatial area or pauses for a designated duration, which is when the followup questions should be asked.

Finding F4: Procedural abstraction and novice clarification queries. Experts often described steps at a high level of abstraction, prompting novices to immediately probe for missing detail (e.g., P2 Novice, in response to an espresso instruction: "How do you do that?"). This indicates that systems must actively detect and resolve this kind of ambiguity rather than assume high-level instructions are sufficient.

Requirement R4: Active Granularity Probing: Because human instructions tend to be high-level, the system must actively probe for granular details, such as specific switch locations or sequence order, when it detects semantic or spatial ambiguity.

4 System Design

4.1 System Design Overview

SPIN captures egocentric audio and video through smart glasses as an author walks through a space and narrates it naturally using speech, gesture, and visual reference. Audio is transcribed and split into segments while corresponding video frames are processed to segment and identify objects; both streams are localized in 3D as the author moves. The transcript drives a language pipeline that extracts semantic attributes and builds a Knowledge Graph, while the video and 3D localization drive a vision pipeline that builds a spatial Scene Graph. The two graphs are then merged into a unified *Spatially Aware Knowledge Graph*.

Because this structure is explicit rather than a raw stream of footage or transcript, gaps are easy to detect automatically: an object with no explanation, or a task missing key steps, surface directly from the graph. *SPIN* uses these gaps to generate targeted follow-up questions posed to the author, whose spoken responses re-enter the same capture-to-graph pipeline, allowing the graph to be iteratively completed over the course of a single walkthrough. The full pipeline is shown in Figure 2.

This design is directly grounded in the formative findings from Section 3.2 (R1–R4), which we discuss in more detail in the following sections.

4.2 Multimodal Spatial Capture System

Motivated by R1 (Multimodal Grounding), our formative study found that experts rely on more than speech alone to communicate spatial knowledge: object references are frequently deictic and only resolvable in conjunction with where the author is looking or pointing at the time. *SPIN* is therefore built around synchronized capture of three streams: audio, video, and derived spatial pose, rather than treating narration as a standalone transcript to be processed after the fact. Audio is transcribed and split into short narration segments, each of which serves as the unit of processing for the rest of the pipeline. Instead of processing every video frame, *SPIN* selects a small number of representative frames per segment, filtered for sharpness and visual diversity, reducing latency and redundancy without sacrificing coverage.

4.3 Spatial Information Capture

To resolve deictic references and object mentions into absolute spatial coordinates, *SPIN* needs to recover not just what the author said and saw, but precisely where they were and what they were looking at when they said it. Ideally, this localization would be performed online without requiring any prior scan of the space, and our architecture supports this in principle. In practice, however, many current consumer smart glasses expose only a single RGB camera with no IMU, and monocular SLAM under these constraints is not yet reliable enough for accurate real-time spatial grounding. We therefore assume access to a pre-existing 3D scan of the space, against which incoming frames are localized to recover

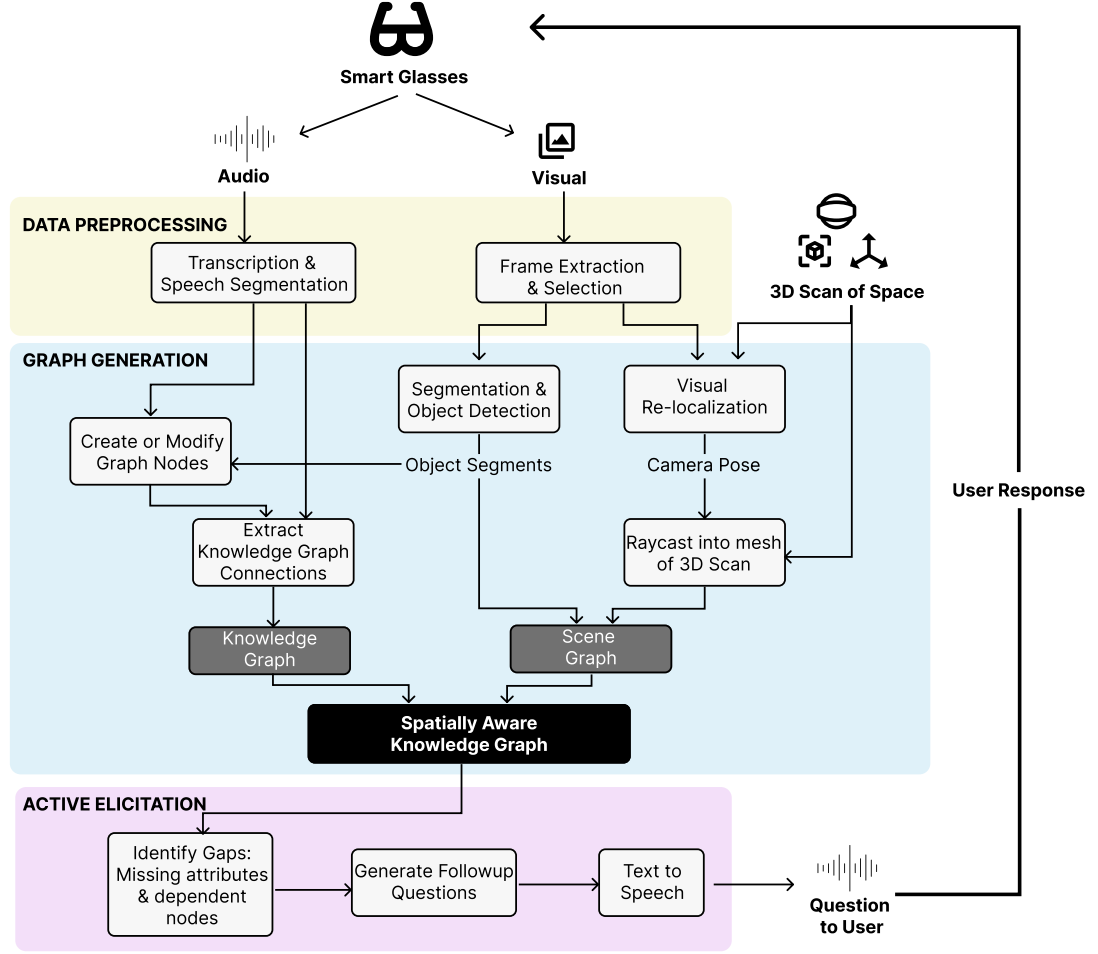


Fig. 2. **System overview of SPIN.** An author wearing smart glasses records synchronized audio and video while walking through a space. During data preprocessing, audio is transcribed and segmented while video frames are extracted and selected. In graph generation, the language pipeline extracts and verifies knowledge graph nodes and edges from the transcript, while the vision pipeline segments and localizes objects against a 3D scan of the space to build a scene graph. The two are merged into a unified Spatially Aware Knowledge Graph. During active elicitation, the system identifies gaps in the graph, such as missing attributes or unexplained objects, generates targeted follow-up questions, and delivers them to the author as speech. The author’s response re-enters the pipeline as new narration, closing the loop.

the camera pose at each moment. This pose is then used to project detected objects from 2D image coordinates into absolute 3D positions, grounding every object mention in a consistent spatial coordinate across the entire walkthrough and forming the geometric backbone of the generated Scene Graph.

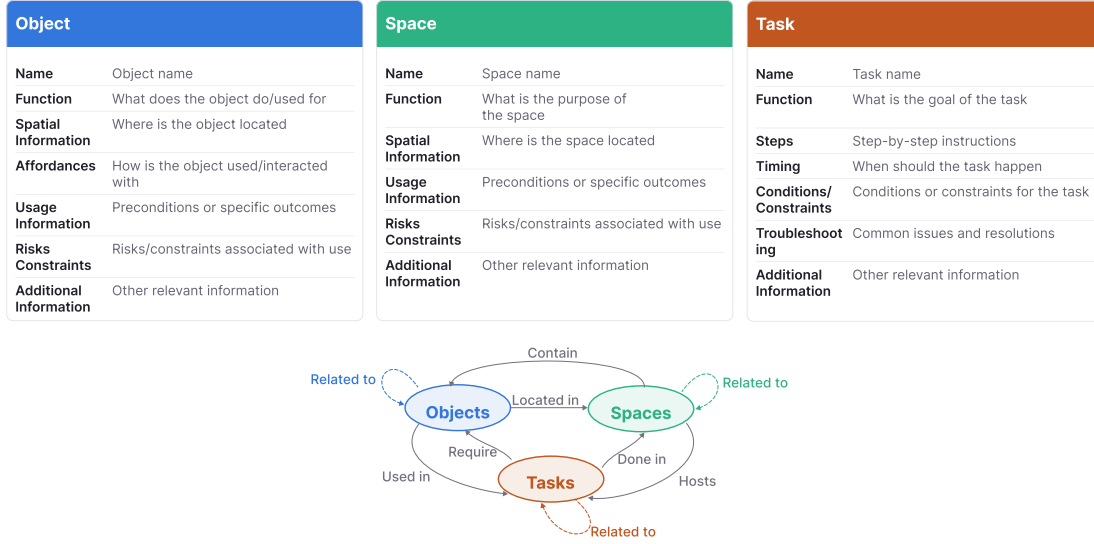


Fig. 3. **The graph schema used by SPIN.** Each node type (Object, Space, and Task) has a defined set of semantic attributes. Relational edges connect nodes across types, encoding functional dependencies such as which objects a task requires, which space a task is done in, and which objects a space contains.

4.4 Spatially Aware Knowledge Graph

At the core of *SPIN* is a structured graph representation that encodes the spatial knowledge captured during a walkthrough as a set of nodes, each with semantic attributes, connected by relational edges. This graph is built from two complementary sources: textual data from the author’s narration, which provides semantic meaning, and visual and spatial data from the egocentric video and 3D localization, which grounds those semantics in the physical space. In the following subsections, we describe the schema that defines the graph’s structure, and the language and vision pipelines that populate it, before describing how the two are merged into a unified Spatially Aware Knowledge Graph.

4.4.1 Graph Schema. *SPIN* represents spatial knowledge using three types of nodes: Objects, Spaces, and Tasks, whose attributes are shown in Figure 3. Object nodes capture properties such as function, spatial location, affordances, usage information, and risks or edge cases. Space nodes describe the purpose, location, and constraints of a given area. Task nodes encode goal, step-by-step instructions, timing, conditions, and common troubleshooting information. This schema was derived from our formative study: by observing the types of information experts communicated during walkthroughs, and what novices needed to successfully complete tasks, we identified Objects, Spaces, and Tasks as the natural categories of spatial knowledge, and defined their attributes accordingly. By organizing information into nodes with structured attributes, *SPIN* can maintain a goal-directed view of what has been covered, what is missing, and how different pieces of information relate to one another.

Nodes across all three types are connected by relational edges. In the Knowledge Graph, these relationships are semantic, encoding functional dependencies such as which objects a task requires, or which tasks a space hosts. In the

Scene Graph, relationships are positional, encoding spatial containment and proximity derived 3D localization as well as visual cues in the images. This distinction between semantic and positional structure is what motivates maintaining two separate graphs before merging them.

4.4.2 Language Pipeline. For each speech segment, *SPIN* extracts the Objects, Spaces, and Tasks referenced in the transcript, instantiating new graph nodes or updating existing ones if the mention refers to something already in the graph. Candidate relational edges between nodes are then proposed by constructing dependency paths from the narration. However, these dependency paths alone are insufficient for more complex or implicit relationships, and passing edge extraction entirely to an LLM tends to over-generate edges beyond what the graph requires. *SPIN* therefore uses a two-stage approach: a lightweight parser proposes candidate edges, and a subsequent LLM call verifies them and assigns typed relationship labels drawn from the schema (such as *Requires*, *Located in*, or *Done in*), discarding candidates that do not reflect a meaningful semantic dependency. At the same time, the semantic attributes for each node are filled in from the transcript content.

Finally, for attributes that are not explicitly stated but are broadly inferable from world knowledge, such as the function of a common household object, *SPIN* allows the LLM to fill these in automatically, unless the author has specified otherwise. This prevents common objects from remaining semantically empty simply because the author did not narrate their obvious purpose, while still deferring to the author’s description when one is provided. Together, these steps produce the Knowledge Graph portion of the Spatially-Aware Knowledge Graph.

4.4.3 Vision Pipeline. For each selected frame, *SPIN* first segments all visible objects in the scene, producing a set of candidate masks covering everything in the author’s field of view. In parallel, *SPIN* detects and segments specifically those objects mentioned in the transcript, grounding language-level entity mentions in their corresponding image regions.

Each resulting mask is encoded into a visual embedding capturing the semantic appearance of the object. All detected objects, whether mentioned in the transcript or not, are retained in the Scene Graph to maintain a complete spatial inventory of the space. For objects that match a language-side node, their embeddings are compared against those of existing graph nodes to support re-identification: determining whether a detected object has been seen before or is a new instance. This embedding-based matching is more robust than position or label matching alone, as it handles cases where the same object appears from a different angle or under different lighting, with spatial pose further assisting disambiguation when two visually similar objects are far apart in 3D space. These matched objects form the bridge between the Scene Graph and the Knowledge Graph.

Finally, detected masks are labeled and their spatial relationships within the image identified, such as which objects are adjacent, contained within, or on top of one another. Combined with the 3D positions recovered from localization, this produces the Scene Graph: a geometric representation in which each node corresponds to a detected object with its segmentation mask, visual embedding, and absolute position within the space.

4.4.4 Merging. Once both the Knowledge Graph and Scene Graph have been constructed for a given segment, *SPIN* merges them into a unified Spatially Aware Knowledge Graph. Nodes matched between the two graphs are merged directly, with any remaining unmatched nodes reconciled by comparing labels across the two graphs to identify overlapping or semantically equivalent entities.

When two nodes are merged, the resulting node retains both the semantic relational edges from the Knowledge Graph and the positional edges from the Scene Graph, giving each merged node both a functional and a spatial identity.

Nodes that appear only in the Scene Graph, objects that were visible but never mentioned, are retained as spatially grounded but semantically empty. Nodes that appear only in the Knowledge Graph, objects that were mentioned but not visually detected, are retained with their semantic attributes but without spatial grounding. Many nodes end up with contributions from both graphs, carrying the full complement of semantic attributes, relational edges, visual embeddings, and 3D position.

This three-way structure is precisely what makes gaps easily detectable: a Scene Graph node with no Knowledge Graph counterpart represents an object the author never explained, and a Knowledge Graph node with empty attributes represents information the author mentioned but did not fully describe.

4.5 Active Elicitation: Closing the Information Gap

With the Spatially-Aware Knowledge Graph constructed, gaps in the authored instructions become directly inspectable. *SPIN* identifies two categories of gaps: missing attributes, where a node exists but one or more schema fields remain unfilled, and missing related nodes, where an object or task that would reasonably be expected given the current graph has not been mentioned. The latter is detected in two ways: by checking Scene Graph nodes that have no Knowledge Graph counterpart, representing objects the author walked past but never explained, and by using an LLM to reason over existing Knowledge Graph nodes and identify related objects that were not mentioned but would likely be needed, for example, a coffee machine node with no associated coffee cup.

From these gaps, *SPIN* generates up to five follow-up questions, with at most one question per node, chosen to be as diverse as possible across the identified gaps. To ensure questions are grounded and consistent, each missing attribute has a corresponding template, for example, a missing function field on a Space node generates the candidate question "What is the purpose of the name?" These templates are provided to the LLM as context to guide question phrasing. Questions that were already asked in a previous round are dropped, and if the same node is identified again, a new question targeting a different gap on that node is generated instead.

Once the author responds, their answer re-enters the full capture-to-graph pipeline, updating the graph before the next round of gap detection and question generation begins.

The full user interface, which contains the streamed egocentric data, the generated graph visualization, session information, generated followup questions, and conversational query panel, is shown in Figure 4.

5 Implementation

SPIN runs on a local machine equipped with an NVIDIA RTX 4090 GPU for all locally-run vision models, with remaining components relying on remote API calls as shown in Figure 5. Capture is performed using Meta Ray-Ban smart glasses paired with a companion smartphone app that streams egocentric audio and video to the server over WebRTC [60], with no on-device computation beyond streaming. Audio is transcribed and segmented using Whisper (whisper-1) [44]; video is streamed at up to 24 fps at 720×1280, though actual throughput is measured at runtime as WebRTC degrades under bandwidth pressure.

Localization is performed using hloc [49] on the frames selected for vision processing, recovering the 6-DOF camera pose of each frame within the pre-scanned mesh. This pose is used to raycast from the camera into the 3D mesh, projecting each detected object’s image location into an absolute 3D position. The language pipeline uses GPT-4o [1] for node extraction and attribute filling, with spaCy [19] proposing candidate edges via syntactic dependency paths and a subsequent GPT-4o call verifying and labeling them. The vision pipeline runs SAM for scene-wide segmentation, GroundingDINO combined with SAM [47] for mentioned-object detection, and DINOv2 [39] for visual embeddings.

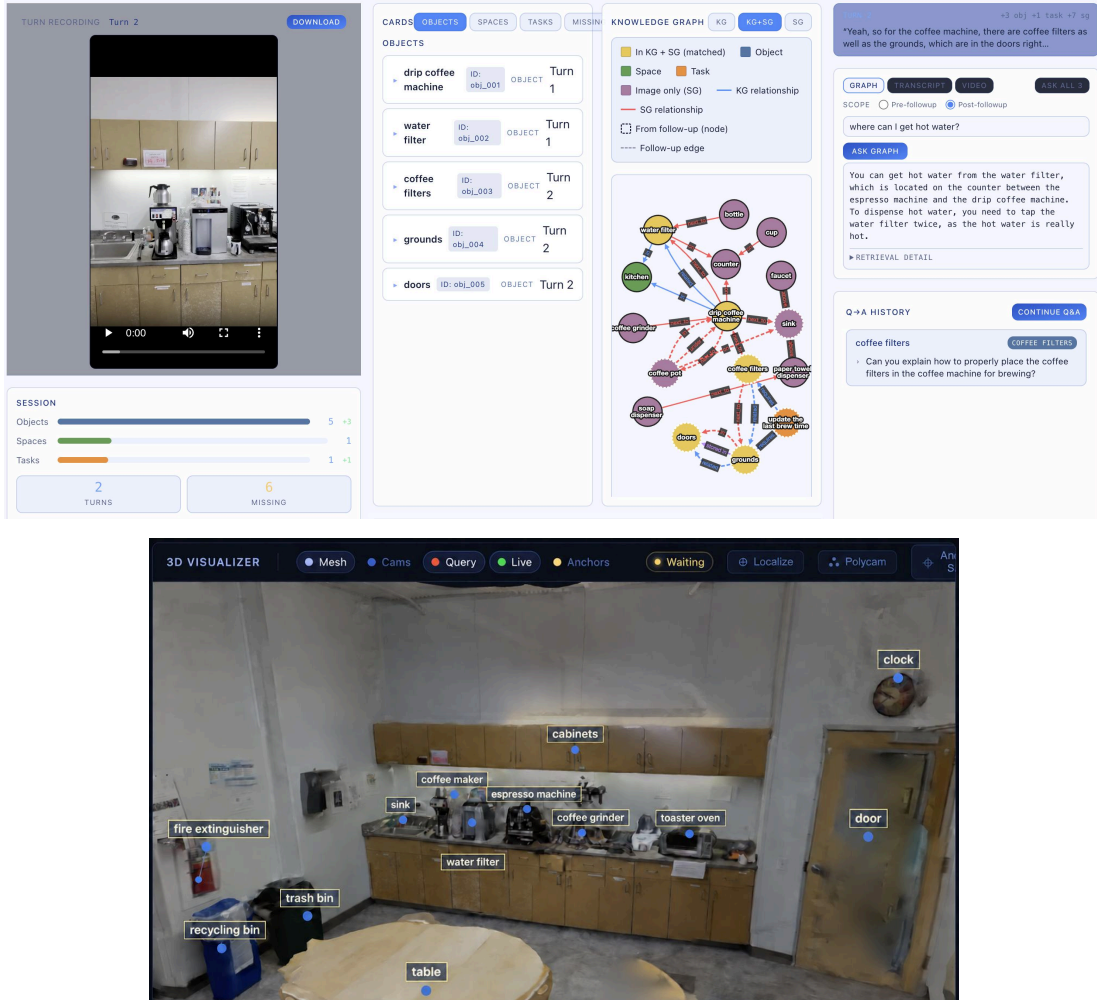


Fig. 4. **The SPIN interface.** (Top) The authoring interface displays the egocentric video feed, the list of captured objects, the generated Spatially Aware Knowledge Graph showing semantic and spatial relationships, and a conversational query panel for querying the graph with natural language. Session statistics show the number of objects, spaces, and tasks captured, as well as gaps remaining. (Bottom) The 3D visualizer overlays detected and localized objects as anchored labels on a pre-scanned mesh of the space, allowing authors to verify that instructions are correctly grounded in their physical context.

Spatial relationships are labeled using GPT-4o, and spatial attributes filled using GPT-4o-mini. Merging uses object name as the primary matching signal, with a lightweight GPT-4o-mini call reconciling any remaining unmatched nodes.

In practice, the latency introduced by the LLM pipeline (ranging from 15–70 seconds per segment depending on input length) made mid-walkthrough delivery of elicitation questions disruptive to the authoring experience. Rather than interrupting the author mid-narration with a question that arrives tens of seconds after the relevant moment has passed, we deliver all follow-up questions at the end of the walkthrough once the author has finished their tour. This preserves the graph’s iterative update structure, where each response re-enters the pipeline and updates the graph

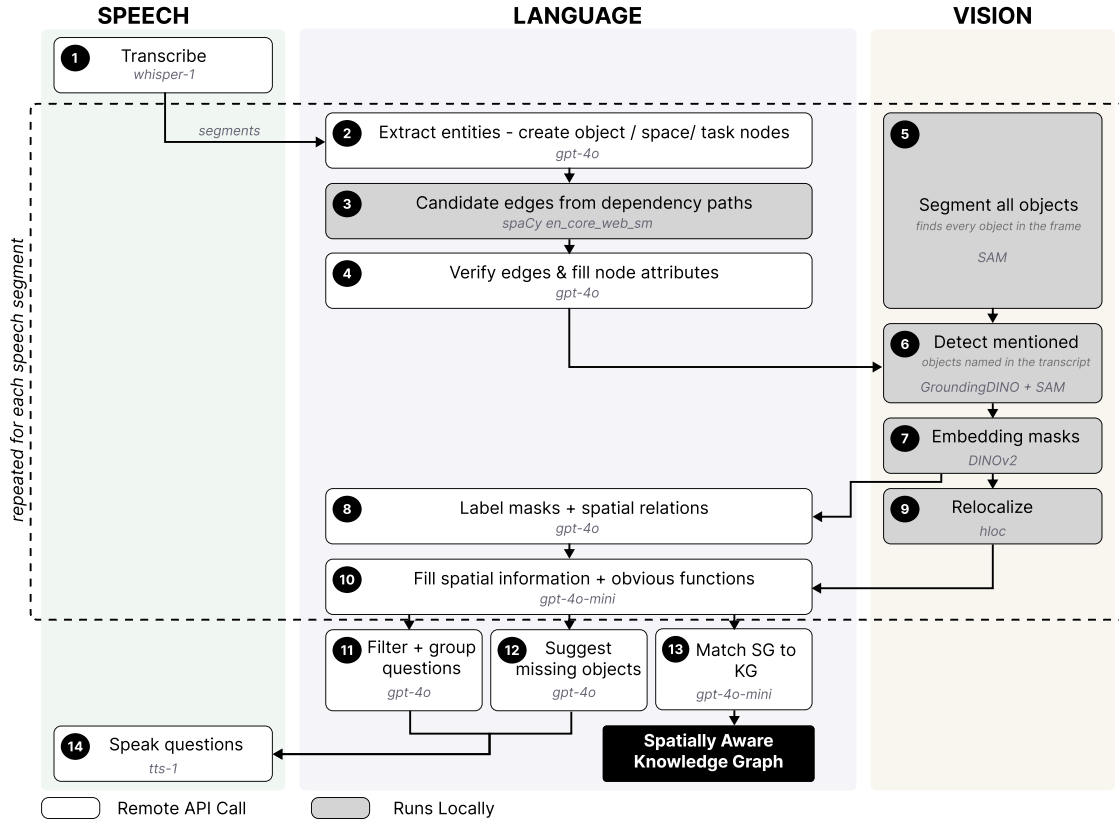


Fig. 5. **Detailed pipeline of SPIN.** Each numbered step corresponds to a model call, shown with the specific model used. Steps 2–10 are repeated for each speech segment. White boxes indicate remote API calls; grey boxes run locally. The language pipeline (steps 1–4) extracts and verifies graph nodes and edges from the transcript. The vision pipeline (steps 5–9) segments, detects, and localizes objects. Steps 10–13 merge the two graphs and identify gaps for active elicitation, which are spoken to the author as follow-up questions (step 14).

before the next round of gap detection begins, while avoiding interruptions during active narration. We note that this is a practical limitation of current LLM inference speeds rather than a fundamental constraint of the approach: as latency decreases, mid-walkthrough delivery becomes increasingly viable.

One practical challenge we encountered was that the glasses’ built-in voice activity detection would occasionally misinterpret background noise or pauses mid-thought as the end of a response. To address this, we added a physical clicker that the author uses to manually mark the start and end of each response, allowing them to pause mid-thought or use the system in noisier environments without triggering a premature cutoff. This is a known limitation of current voice activity detection models, which do not yet reliably distinguish intentional pauses from the end of an utterance.

6 Evaluation

6.1 User Study

To evaluate *SPIN*, we conducted a user study in which domain experts authored spatial instructions for a real physical space using our system. We recruited ten participants (5 male, 5 female, ages 18-65) who were intimately familiar with their assigned space: five were assigned to a research lab and five to a kitchen space. Participants were asked to wear the Meta Ray-Ban glasses connected to our system and give a natural walkthrough of their space, with the instruction to imagine a new person who will be using that space every day and to communicate everything important they would need to know to use it independently, without further help. What they chose to cover and in what order was left entirely up to them. Initial walkthroughs lasted between 5-9 minutes. Upon completing the walkthrough, *SPIN* generated follow-up questions based on the gaps identified in the Spatially Aware Knowledge Graph constructed from the initial walkthrough. Participants then answered these questions while still wearing the glasses, allowing the system to capture both their verbal responses and visual context. This elicitation phase lasted an additional 8-10 minutes depending on the level of detail provided and the number of questions generated. This design allowed us to capture a snapshot of the knowledge graph both before elicitation (Condition A: walk-and-talk only) and after (Condition B: walk-and-talk with follow-up questions). All participants were seeing the system for the first time, with no prior training required. This study received exempt status from our Institutional Review Board.³

6.1.1 Information Completeness. To determine whether the two environments (kitchen, N=5; lab, N=5) could be analyzed as a single sample, we first ran a one-way ANOVA with location as a between-subjects factor across all three metrics (node count, edge count, and attribute coverage) at baseline, post-elicitation, and delta, yielding nine comparisons in total. None of the comparisons reached significance ($F \leq 2.21$, all $p \geq .05$), so we pooled the two groups into a single sample of N=10 for all subsequent analyses.

As shown in Figure 6, active elicitation produced significant increases across all three measures of graph completeness. The number of entities identified increased from a mean of 16.90 to 24.50 ($Z = 2.80$, $p = .002$, $r = .89$), and the number of relationships captured increased from 12.20 to 26.20 ($Z = 2.80$, $p = .002$, $r = .89$), representing a mean gain of 14 additional relational edges per walkthrough. Attribute coverage increased from 51.2% to 64.1% ($Z = 2.59$, $p = .006$, $r = .82$), indicating that the elicitation phase filled in a meaningful portion of schema attributes that authors did not provide during the initial walkthrough alone. Together, these results suggest that active elicitation reliably surfaces information that authors omit during unassisted walkthroughs, producing a more complete knowledge graph without requiring additional unguided effort from the author. Notably, these gains were achieved with a cap of 15 follow-up questions per session; we anticipate that allowing additional rounds of elicitation would increase coverage further, particularly for attribute completeness.

6.1.2 Search Retrieval Accuracy. To evaluate the quality and usability of the generated graphs, we constructed a benchmark of 20 questions per space based on the tasks and objects present in the lab and kitchen environments. Questions were created before any participant data was collected, with no knowledge of what participants would choose to cover in their walkthroughs. The questions are equally distributed across five categories: Direct Visual Search, Functional and Description Search, Task-Specific Details, Usage and Operational Logic, and Edge Cases and Constraints. The full question set is provided in the Appendix A.

³IRB details anonymized for review.

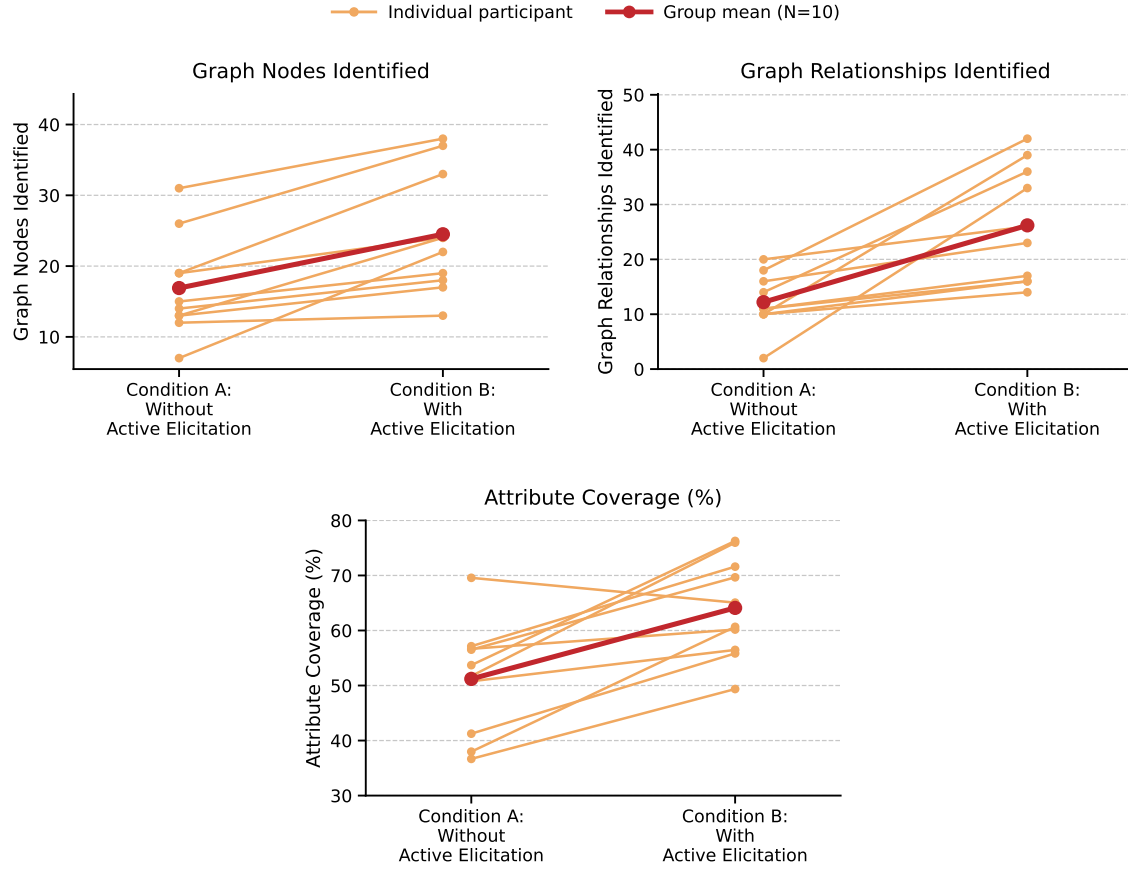


Fig. 6. Information completeness before and after active elicitation across all ten participants (N=10, pooled across kitchen and lab spaces). Each orange line represents an individual participant; the red line shows the group mean. Wilcoxon signed-rank tests confirmed significant increases across all three metrics (node count: $Z = 2.80$, $p = .002$, $r = .89$, edge count: $Z = 2.80$, $p = .002$, $r = .89$, attribute coverage: $Z = 2.59$, $p = .006$, $r = .82$).

For each participant’s generated data, an LLM was prompted to answer each benchmark question using only the information available in that condition. Answers were scored on a 0-3 scale: 0 indicates no relevant information or a hallucinated answer, 1 indicates the entity was named but with no supporting detail, 2 indicates a partial answer, and 3 indicates a full, grounded response. Two authors each independently scored half of the responses, with the remaining half double-coded to establish inter-rater reliability. Agreement was very high ($\kappa = 0.958$, quadratic-weighted Cohen’s kappa [9]), and scores from the double-coded items were averaged across raters before analysis.

We evaluated four conditions: **SPIN** (with active elicitation), **SPIN** without active elicitation, Video + VLM (Gemini via NotebookLM [13], queried directly from the egocentric video), and Transcript + LLM (GPT-4o, queried from the raw speech transcript). The video and transcript baselines are evaluated without elicitation, as the active elicitation phase is built from the graph representation and cannot be meaningfully applied to unstructured video or transcript inputs.

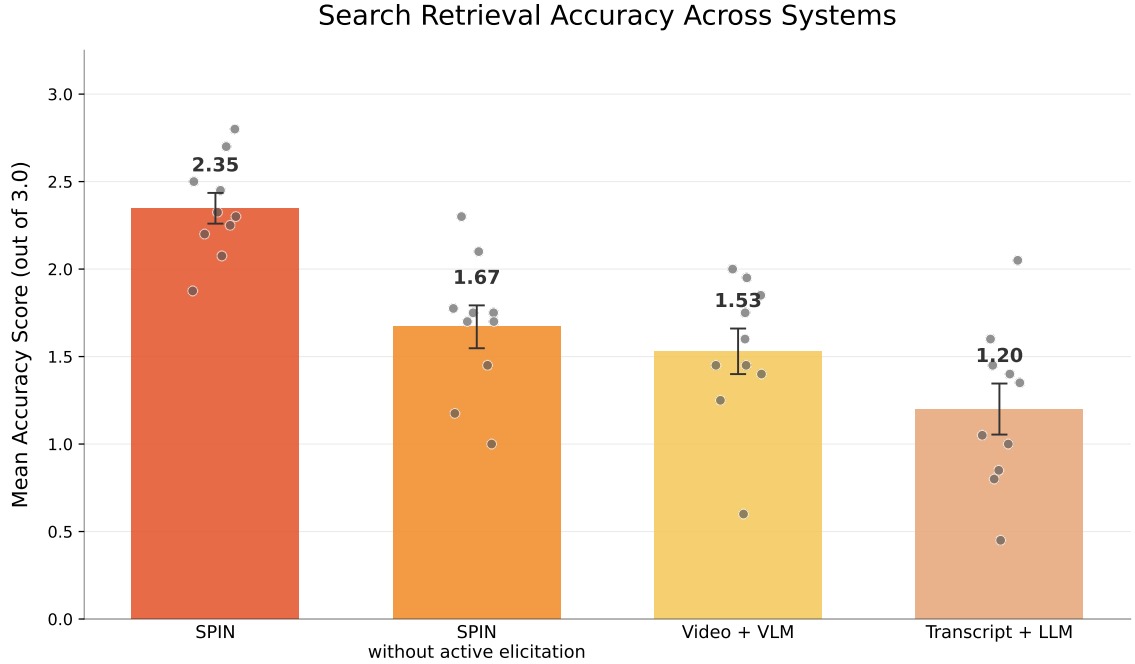


Fig. 7. Mean search retrieval accuracy scores (0-3) across four conditions (N=10 per condition). SPIN significantly outperforms all baselines ($Z = 2.80$, $p = .002$, $r = .89$, Wilcoxon signed-rank). Individual participant means are shown as grey dots; error bars indicate standard error of the mean. The video and transcript baselines are evaluated without active elicitation, as elicitation is intrinsic to the graph representation.

As shown in Figure 7, **SPIN** achieved the highest mean accuracy score of 2.35 out of 3.0, significantly outperforming all three comparisons (all $Z = 2.80$, $p = .002$, $r = .89$). SPIN without active elicitation scored 1.67, Video + VLM scored 1.53, and Transcript + LLM scored 1.20. The gap between SPIN with and without active elicitation ($\Delta = 0.68$) demonstrates the contribution of the elicitation phase beyond graph construction alone. The gap between SPIN without active elicitation and the video and transcript baselines further demonstrates the value of the structured graph representation itself, independent of elicitation, for supporting accurate and grounded question answering.

6.1.3 Qualitative usability metrics. Participants rated SPIN highly on standard usability measures. The System Usability Scale score of 86.5 falls in the *excellent* range, well above the industry benchmark of 68, and the NASA-TLX workload score of 28.0 indicates low perceived task load. Together these suggest that the system was both easy to use and not cognitively demanding despite the novelty of the interaction. The full post study questionnaire can be seen in the Appendix B

Figure 8 shows responses to our nine custom post-task items, grouped into *instruction quality and confidence* (items 1-4) and *active elicitation and interface* (items 5-9). Across the instruction quality items, participants agreed that the system supported explaining spatial relationships effectively (3.75/5) and that a newcomer could follow the resulting instructions (3.62/5). Ratings on active elicitation items were notably positive: participants agreed that follow-up questions helped them recall more details (4.20/5) and served as a helpful guide (4.40/5). Notably, the Interruption item received the lowest mean score (2.30/5), indicating that participants largely did not find the elicitation disruptive.

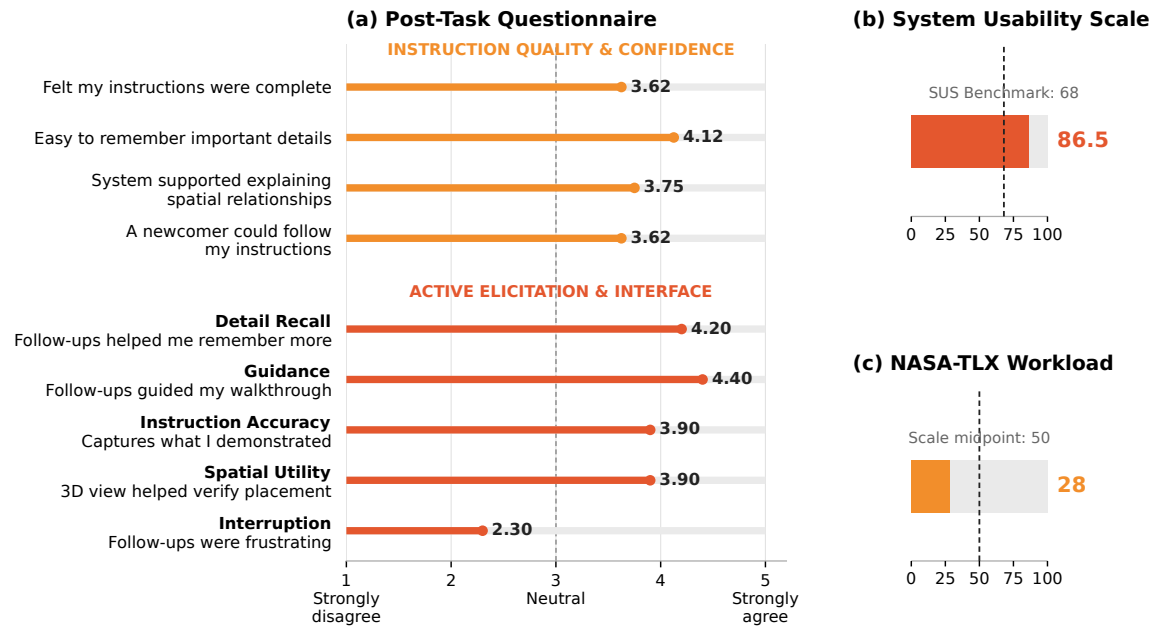


Fig. 8. Post-task questionnaire results for N=10 participants. (a) Custom Likert items grouped into instruction quality and confidence, and active elicitation and interface, scored out of 5. (b) System Usability Scale score of 86.5, well above the industry benchmark of 68. (c) NASA-TLX workload score of 28.0, indicating low perceived task load.

Qualitative responses reinforced these findings. Participants valued the hands-free, natural authoring experience: “*I have access to both my hands! I just use natural language. I do not have to sit down and type to explain my thinking.*” The follow-up questions were frequently cited as surfacing details participants had not thought to include: *The follow up question about rules around using the shared refrigerator was a very useful question, and I forgot to talk about it during my initial walk-around,* and “*The followup on ssh/gpu access and usage of 3d cameras were brought up by the system which I forgot to mention.*” All participants responded to the expert blindness question that the system prompted them to recall at least one omitted detail. Participants also identified areas for improvement, noting that the system occasionally misheard words or asked questions that felt overly literal, and expressing a desire for more flexibility to proactively edit instructions between rounds.

6.2 System Benchmarks

We report the processing latency of the SPIN pipeline as a function of input length, since processing time scales with the duration of the narration segment rather than being fixed. For a short input clip of approximately 20 seconds, the pipeline completes in roughly 22-26 seconds of processing time. For a longer input clip of approximately 90 seconds, processing takes approximately 95 seconds. Table 1 shows the breakdown of processing time across pipeline stages. The dominant cost is per-segment processing, which accounts for approximately 75% of total processing time, driven primarily by visual re-localization via hloc (~30% of total). Local vision models (SAM, GroundingDINO, DINOv2, hloc) collectively require approximately 15.3 GB of GPU VRAM on an NVIDIA RTX 4090 (24 GB total). These latency

Table 1. Pipeline stage breakdown by average percentage of total processing time. Processing time scales with input length: a ~20s narration segment takes approximately 22–26s to process, while a ~90s segment takes approximately 95s.

Pipeline Stage	Avg. % of Processing Time
Segment Loop (per-segment processing)	~75%
Relocalize / hloc (step 9)	~30%
Transcribe / Whisper (step 1)	~8%
Filter + Group Questions (step 11–13)	~8%
Match SG to KG (step 13)	~9%

characteristics motivated our decision to deliver elicitation questions at the end of the walkthrough rather than between segments.

7 Discussion

Lowering the barrier to spatial instruction authoring. *SPIN* demonstrates that spatial instructions can be captured through a modality that already comes naturally to people: walking through a space and talking about it, mimicking how they would give instructions to another person physically present there. By grounding narration in egocentric video and 3D localization, the system removes the need for authors to manually document steps, annotate images, or structure content ahead of time. The resulting graph is not only easier to author than existing approaches, but opens up a range of downstream applications, from conversational querying to AR overlays to exportable manuals, that would be difficult to support from unstructured video or text alone.

Toward real-time elicitation. A deliberate design goal of *SPIN* was to deliver follow-up questions at natural transition points during the walkthrough, so that the author is still physically situated near the relevant object or space when they respond. In practice, LLM pipeline latency made this infeasible, and we moved question delivery to the end of the walkthrough. While this still produced meaningful gains in completeness, we believe mid-walkthrough elicitation would be more effective: responses given in context are likely to be richer and more grounded than those given in retrospect. As inference speeds continue to improve, this becomes an increasingly viable direction.

Balancing LLM priors with author intent. A challenge in designing *SPIN* is deciding how much to rely on what the author explicitly provides versus what the LLM can reasonably infer. We deliberately kept LLM-inferred content narrow, limited to broadly obvious object functions, and deferred to the author’s description whenever one was provided. In early iterations, allowing the LLM more latitude produced graphs that were superficially complete but included assumptions the author had not made and might not endorse. The right balance between author fidelity and LLM augmentation is likely task-dependent: downstream applications that prioritize completeness may tolerate more inference, while those that require precision may need to restrict it further.

The role of task scope in elicitation quality. We observed that the specificity of the authoring context meaningfully affects the quality of follow-up questions generated. When the system has a clear sense of the tasks and objects relevant to a space, it can generate targeted, useful questions. However, framing the task too narrowly risks missing general-purpose information that might be useful for later tasks, while framing it too broadly can produce questions that lack the depth needed to be actionable. Future work should explore how to dynamically calibrate elicitation scope based on the downstream use case, rather than applying a fixed strategy across all spaces and tasks.

Beyond entity-level representation. The current graph schema organizes knowledge at the level of Objects, Spaces, and Tasks, which proved sufficient for the instruction authoring and retrieval tasks we evaluated. However, there is likely a richer layer of higher-order reasoning that this structure does not yet capture: causal dependencies between tasks, conditional instructions that vary by user or context, or abstract routines that span multiple spaces. Importantly, this knowledge can be very contextual, as the same space means different things to different people, and instructions that feel complete to one author may miss what another considers essential. Future work could explore how to incorporate multiple perspectives on a space into a unified graph, or how to personalize instruction retrieval based on who is asking and why. The instruction set itself could also be enriched beyond what the author narrates. The egocentric video contains richer signal than static frame sampling captures, including physical demonstrations and activity that carry instructional meaning, and external sources such as appliance manuals or internet blogposts could further augment what the graph knows about a space.

8 Conclusion

This paper presents *SPIN*, a spatial instruction authoring system that captures multimodal egocentric data to generate grounded, spatially-aware instructions from a natural walkthrough. At its core is the Spatially Aware Knowledge Graph, a structured representation that links verbal content to visual context and physical location, enabling both efficient retrieval and automatic gap detection. Building on this, *SPIN*'s active elicitation pipeline identifies missing information and poses targeted follow-up questions to the author, improving the completeness of the resulting instruction set without additional unguided effort. Together, these components offer a new approach to spatial instruction authoring that is lower burden, more complete, and more grounded than existing methods.

References

- [1] 2024. OpenAI GPT-4o. <https://openai.com/index/hello-gpt-4o>. Online. Accessed: July 2024.
- [2] Iro Armeni, Zhi-Yang He, JunYoung Gwak, Amir R. Zamir, Martin Fischer, Jitendra Malik, and Silvio Savarese. 2019. 3D Scene Graph: A Structure for Unified Semantics, 3D Space, and Camera. arXiv:1910.02527 [cs.CV] <https://arxiv.org/abs/1910.02527>
- [3] Andrea Bellucci, Alberto Ruiz, Paloma Díaz, and Igancio Aedo. 2018. Investigating augmented reality support for novice users in circuit prototyping. In *Proceedings of the 2018 international conference on advanced visual interfaces*. 1–5.
- [4] Bhaskar Bhattacharya and Eliot H. Winer. 2019. Augmented reality via expert demonstration authoring (AREDA). *Computers in Industry* 105 (2019), 61–79. doi:10.1016/j.compind.2018.04.021
- [5] Yining Cao, Peiling Jiang, and Haijun Xia. 2025. Generative and Malleable User Interfaces with Generative and Evolving Task-Driven Data Model. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 686, 20 pages. doi:10.1145/3706598.3713285
- [6] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. 2021. Emerging Properties in Self-Supervised Vision Transformers. arXiv:2104.14294 [cs.CV] <https://arxiv.org/abs/2104.14294>
- [7] Chen Chen, Cuong Nguyen, Jane Hoffswell, Jennifer Healey, Trung Bui, and Nadir Weibel. 2023. PaperToPlace: Transforming Instruction Documents into Spatialized and Context-Aware Mixed Reality Experiences. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) (*UIST '23*). Association for Computing Machinery, New York, NY, USA, Article 118, 21 pages. doi:10.1145/3586183.3606832
- [8] Subramanian Chidambaram, Hank Huang, Fengming He, Xun Qian, Ana M Villanueva, Thomas S Redick, Wolfgang Stuerzlinger, and Karthik Ramani. 2021. ProcessAR: An augmented reality-based tool to create in-situ procedural 2D/3D AR Instructions. In *Proceedings of the 2021 ACM Designing Interactive Systems Conference* (Virtual Event, USA) (*DIS '21*). Association for Computing Machinery, New York, NY, USA, 234–249. doi:10.1145/3461778.3462126
- [9] Jacob Cohen. 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement* 20 (1960), 37 – 46. <https://api.semanticscholar.org/CorpusID:15926286>
- [10] Mustafa Doga Dogan, Eric J Gonzalez, Karan Ahuja, Ruofei Du, Andrea Colaço, Johnny Lee, Mar Gonzalez-Franco, and David Kim. 2024. Augmented Object Intelligence with XR-Objects. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology* (Pittsburgh, PA, USA) (*UIST '24*). Association for Computing Machinery, New York, NY, USA, Article 19, 15 pages. doi:10.1145/3654777.3676379
- [11] Gemini Team et al. 2025. Gemini: A Family of Highly Capable Multimodal Models. arXiv:2312.11805 [cs.CL] <https://arxiv.org/abs/2312.11805>

- [12] Marius Fischer, Bernhard Fuerst, Sing Chun Lee, Javad Fotouhi, Severine Habert, Simon Weidert, Ekkehard Euler, Greg Osgood, and Nassir Navab. 2016. Preclinical usability study of multiple augmented reality concepts for K-wire placement. *International journal of computer assisted radiology and surgery* 11, 6 (2016), 1007–1014.
- [13] Google. 2026. NotebookLM. <https://notebooklm.google.com>. An AI-powered research assistant and note-taking tool.
- [14] Huy Ha and Shuran Song. 2022. Semantic Abstraction: Open-World 3D Scene Understanding from 2D Vision-Language Models. arXiv:2207.11514 [cs.CV] <https://arxiv.org/abs/2207.11514>
- [15] Haoyu Han, Yu Wang, Harry Shomer, Kai Guo, Jiayuan Ding, Yongjia Lei, Mahantesh Halappanavar, Ryan A. Rossi, Subhabrata Mukherjee, Xianfeng Tang, Qi He, Zhigang Hua, Bo Long, Tong Zhao, Neil Shah, Amin Javari, Yinglong Xia, and Jiliang Tang. 2025. Retrieval-Augmented Generation with Graphs (GraphRAG). arXiv:2501.00309 [cs.IR] <https://arxiv.org/abs/2501.00309>
- [16] Steven J. Henderson and Steven Feiner. 2009. Evaluating the benefits of augmented reality for task localization in maintenance of an armored personnel carrier turret. In *2009 8th IEEE International Symposium on Mixed and Augmented Reality*. 135–144. doi:10.1109/ISMAR.2009.5336486
- [17] Pamela Hinds. 1999. The Curse of Expertise: The Effects of Expertise and Debiasing Methods on Predictions of Novice Performance. *Journal of Experimental Psychology: Applied* 5 (06 1999), 205–221. doi:10.1037/1076-898X.5.2.205
- [18] Yining Hong, Chunru Lin, Yilun Du, Zhenfang Chen, Joshua B. Tenenbaum, and Chuang Gan. 2023. 3D Concept Learning and Reasoning from Multi-View Images. arXiv:2303.11327 [cs.CV] <https://arxiv.org/abs/2303.11327>
- [19] Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python. (2020). doi:10.5281/zenodo.1212303
- [20] Eric Horvitz. 1999. Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Pittsburgh, Pennsylvania, USA) (CHI '99). Association for Computing Machinery, New York, NY, USA, 159–166. doi:10.1145/302979.303030
- [21] Nathan Hughes, Yun Chang, and Luca Carlone. 2022. Hydra: A Real-time Spatial Perception System for 3D Scene Graph Construction and Optimization. arXiv:2201.13360 [cs.RO] <https://arxiv.org/abs/2201.13360>
- [22] Mina Huh, Zihui Xue, Ujjaini Das, Kumar Ashutosh, Kristen Grauman, and Amy Pavel. 2025. Vid2Coach: Transforming How-To Videos into Task Assistants. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25)*. Association for Computing Machinery, New York, NY, USA, Article 46, 24 pages. doi:10.1145/3746059.3747612
- [23] Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and Philip S. Yu. 2022. A Survey on Knowledge Graphs: Representation, Acquisition, and Applications. *IEEE Transactions on Neural Networks and Learning Systems* 33, 2 (Feb. 2022), 494–514. doi:10.1109/tnnls.2021.3070843
- [24] Panayu Keelawat and Ryo Suzuki. 2024. Transforming Procedural Instructions into In-Situ Augmented Reality Guides with InstructAR. In *Adjunct Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology* (Pittsburgh, PA, USA) (UIST Adjunct '24). Association for Computing Machinery, New York, NY, USA, Article 70, 3 pages. doi:10.1145/3672539.3686321
- [25] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. 2023. LERF: Language Embedded Radiance Fields. arXiv:2303.09553 [cs.CV] <https://arxiv.org/abs/2303.09553>
- [26] Junhan Kong, Dena Sabha, Jeffrey P Bigham, Amy Pavel, and Anhong Guo. 2021. TutorialLens: Authoring Interactive Augmented Reality Tutorials Through Narration and Demonstration. In *Proceedings of the 2021 ACM Symposium on Spatial User Interaction* (Virtual Event, USA) (SUI '21). Association for Computing Machinery, New York, NY, USA, Article 16, 11 pages. doi:10.1145/3485279.3485289
- [27] Ryosuke Korekata, Quanting Xie, Yonatan Bisk, and Komei Sugiura. 2026. Affordance RAG: Hierarchical Multimodal Retrieval With Affordance-Aware Embodied Memory for Mobile Manipulation. *IEEE Robotics and Automation Letters* 11, 3 (2026), 2706–2713. doi:10.1109/LRA.2026.3653281
- [28] Todd Kulesza, Margaret Burnett, Weng-Keen Wong, and Simone Stumpf. 2015. Principles of Explanatory Debugging to Personalize Interactive Machine Learning. In *Proceedings of the 20th International Conference on Intelligent User Interfaces* (Atlanta, Georgia, USA) (IUI '15). Association for Computing Machinery, New York, NY, USA, 126–137. doi:10.1145/2678025.2701399
- [29] Chenyi Li, Guande Wu, Gromit Yeuk-Yin Chan, Dishita G Turakhia, Sonia Castelo Quispe, Dong Li, Leslie Welch, Claudio Silva, and Jing Qian. 2025. Satori: Towards Proactive AR Assistant with Belief-Desire-Intention User Modeling. arXiv:2410.16668 [cs.HC] <https://arxiv.org/abs/2410.16668>
- [30] Haichang Li, Qinshi Zhang, Piaohong Wang, and Zhicong Lu. 2026. Mixed-Initiative Context: Structuring and Managing Context for Human-AI Collaboration. arXiv:2604.07121 [cs.HC] <https://arxiv.org/abs/2604.07121>
- [31] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*. PMLR, 19730–19742.
- [32] Yao Li, Gijs Luijten, Christina Gsaxner, Kim Grunert, Alexis Bader, Frank Hölzle, Rainer Röhrig, Matías de la Fuente, Jan Egger, Kunpeng Xie, et al. 2026. Usability Study of Augmented Reality Visualization Modalities on Localization Accuracy in the Head and Neck: Randomized Crossover Trial. *JMIR Serious Games* 14, 1 (2026), e75962.
- [33] Ziyi Liu, Zhengzhe Zhu, Enze Jiang, Feichi Huang, Ana M Villanueva, Xun Qian, Tianyi Wang, and Karthik Ramani. 2023. InstruMentAR: Auto-Generation of Augmented Reality Tutorials for Operating Digital Instruments Through Recording Embodied Demonstration. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) (CHI '23). Association for Computing Machinery, New York, NY, USA, Article 32, 17 pages. doi:10.1145/3544548.3581442
- [34] Michael R. Marner, Andrew Irlitti, and Bruce H. Thomas. 2013. Improving procedural task performance with Augmented Reality annotations. In *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. 39–48. doi:10.1109/ISMAR.2013.6671762
- [35] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. arXiv:2003.08934 [cs.CV] <https://arxiv.org/abs/2003.08934>

- [36] Pranav Mistry and Pattie Maes. 2008. Quickies: Intelligent sticky notes. In *2008 IET 4th International Conference on Intelligent Environments*. 1–4. doi:10.1049/cp:20081185
- [37] Peter Mohr, Bernhard Kerbl, Michael Donoser, Dieter Schmalstieg, and Denis Kalkofen. 2015. Retargeting Technical Documentation to Augmented Reality. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) (*CHI '15*). Association for Computing Machinery, New York, NY, USA, 3337–3346. doi:10.1145/2702123.2702490
- [38] Peter Mohr, David Mandl, Markus Tatzgern, Eduardo Veas, Dieter Schmalstieg, and Denis Kalkofen. 2017. Retargeting Video Tutorials Showing Tools With Surface Contact to Augmented Reality. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) (*CHI '17*). Association for Computing Machinery, New York, NY, USA, 6547–6558. doi:10.1145/3025453.3025688
- [39] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. 2024. DINOv2: Learning Robust Visual Features without Supervision. arXiv:2304.07193 [cs.CV] <https://arxiv.org/abs/2304.07193>
- [40] Riccardo Palmarini, Iñigo Fernández Del Amo, Dedy Ariansyah, Samir Khan, John Ahmet Erkoyuncu, and Rajkumar Roy. 2023. Fast Augmented Reality Authoring: Fast Creation of AR Step-by-Step Procedures for Maintenance Operations. *IEEE Access* 11 (2023), 8407–8421. doi:10.1109/ACCESS.2023.3235871
- [41] Per Persson, Fredrik Espinoza, Petra Fagerberg, Anna Sandin, and Rickard Cöster. 2003. *GeoNotes: A Location-Based Information System for Public Spaces*. Springer London, London, 151–173. doi:10.1007/978-1-4471-0035-5_7
- [42] Zhangyang Qi, Zhixiong Zhang, Ye Fang, Jiaqi Wang, and Hengshuang Zhao. 2025. GPT4Scene: Understand 3D Scenes from Videos with Vision-Language Models. arXiv:2501.01428 [cs.CV] <https://arxiv.org/abs/2501.01428>
- [43] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. arXiv:2103.00020 [cs.CV] <https://arxiv.org/abs/2103.00020>
- [44] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust Speech Recognition via Large-Scale Weak Supervision. arXiv:2212.04356 [eess.AS] <https://arxiv.org/abs/2212.04356>
- [45] Rafael Radkowski. 2015. Investigation of visual features for augmented reality assembly assistance. In *International conference on virtual, augmented and mixed reality*. Springer, 488–498.
- [46] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. 2024. SAM 2: Segment Anything in Images and Videos. arXiv:2408.00714 [cs.CV] <https://arxiv.org/abs/2408.00714>
- [47] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. 2024. Grounded SAM: Assembling Open-World Models for Diverse Visual Tasks. arXiv:2401.14159 [cs.CV] <https://arxiv.org/abs/2401.14159>
- [48] Antoni Rosinol, Marcus Abate, Yun Chang, and Luca Carlone. 2020. Kimera: an Open-Source Library for Real-Time Metric-Semantic Localization and Mapping. arXiv:1910.02490 [cs.RO] <https://arxiv.org/abs/1910.02490>
- [49] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. 2019. From Coarse to Fine: Robust Hierarchical Localization at Large Scale. arXiv:1812.03506 [cs.CV] <https://arxiv.org/abs/1812.03506>
- [50] Saumya Saxena, Blake Buchanan, Chris Paxton, Peiqi Liu, Bingqing Chen, Narunas Vaskevicius, Luigi Palmieri, Jonathan Francis, and Oliver Kroemer. 2025. GraphEQA: Using 3D Semantic Scene Graphs for Real-time Embodied Question Answering. arXiv:2412.14480 [cs.RO] <https://arxiv.org/abs/2412.14480>
- [51] Johannes L. Schönberger and Jan-Michael Frahm. 2016. Structure-from-Motion Revisited. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 4104–4113. doi:10.1109/CVPR.2016.445
- [52] Dhruv Shah, Blazej Osinski, Brian Ichter, and Sergey Levine. 2022. LM-Nav: Robotic Navigation with Large Pre-Trained Models of Language, Vision, and Action. arXiv:2207.04429 [cs.RO] <https://arxiv.org/abs/2207.04429>
- [53] Jingyu Shi, Rahul Jain, Seunggeun Chi, Hyungjun Doh, Hyung-gun Chi, Alexander J. Quinn, and Karthik Ramani. 2025. CARING-AI: Towards Authoring Context-aware Augmented Reality INstruction through Generative Artificial Intelligence. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 29, 23 pages. doi:10.1145/3706598.3713348
- [54] Lucchas Ribeiro Skreinig, Peter Mohr, Blanca Berger, Markus Tatzgern, Dieter Schmalstieg, and Denis Kalkofen. 2024. Immersive Authoring by Demonstration of Industrial Procedures. In *2024 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. 1293–1302. doi:10.1109/ISMAR62088.2024.00146
- [55] Sruti Srinidhi, Edward Lu, and Anthony Rowe. 2024. XaiR: An XR Platform that Integrates Large Language Models with the Physical World. In *2024 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. 759–767. doi:10.1109/ISMAR62088.2024.00091
- [56] Daniel Stover and Doug Bowman. 2024. TAGGAR: General-Purpose Task Guidance from Natural Language in Augmented Reality using Vision-Language Models. In *Proceedings of the 2024 ACM Symposium on Spatial User Interaction* (Trier, Germany) (*SUI '24*). Association for Computing Machinery, New York, NY, USA, Article 12, 12 pages. doi:10.1145/3677386.3682095

- [57] Yusi Sun, Ying Jiang, Jiayin Lu, Yin yang, Yong-Hong Kuo, and Chenfanfu Jiang. 2026. JARVIS: A Just-in-Time AR Visual Instruction System for Cross-Reality Task Guidance. arXiv:2604.10108 [cs.HC] <https://arxiv.org/abs/2604.10108>
- [58] Arthur Tang, Charles Owen, Frank Biocca, and Weimin Mou. 2003. Comparative effectiveness of augmented reality in object assembly. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Ft. Lauderdale, Florida, USA) (*CHI '03*). Association for Computing Machinery, New York, NY, USA, 73–80. doi:10.1145/642611.642626
- [59] Anh Thai, Songyou Peng, Kyle Genova, Leonidas Guibas, and Thomas Funkhouser. 2025. SplatTalk: 3D VQA with Gaussian Splatting. arXiv:2503.06271 [cs.CV] <https://arxiv.org/abs/2503.06271>
- [60] WebRTC Working Group. 2011. WebRTC. <https://webrtc.org/> Online. Accessed: April 2023.
- [61] Giles Westerfield, Antonija Mitrovic, and Mark Billinghurst. 2015. Intelligent augmented reality training for motherboard assembly. *International Journal of Artificial Intelligence in Education* 25, 1 (2015), 157–172.
- [62] Dennis Wixon, Karen Holtzblatt, and Stephen Knox. 1990. Contextual design: an emergent view of system design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Seattle, Washington, USA) (*CHI '90*). Association for Computing Machinery, New York, NY, USA, 329–336. doi:10.1145/97243.97304
- [63] J. Zauner, M. Haller, A. Brandl, and W. Hartman. 2003. Authoring of a mixed reality assembly instructor for hierarchical structures. In *The Second IEEE and ACM International Symposium on Mixed and Augmented Reality, 2003. Proceedings.* 237–246. doi:10.1109/ISMAR.2003.1240707
- [64] Ada Yi Zhao, Aditya Gunturu, Ellen Yi-Luen Do, and Ryo Suzuki. 2025. Guided Reality: Generating Visually-Enriched AR Task Guidance with LLMs and Vision Models. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25)*. Association for Computing Machinery, New York, NY, USA, Article 146, 15 pages. doi:10.1145/3746059.3747784

A Benchmark Question Set

To evaluate the quality and usability of the generated knowledge graphs, we constructed a benchmark of 20 questions per space (40 total across the kitchen and lab environments). Questions were written before any participant data was collected, with no knowledge of what authors would choose to cover in their walkthroughs, ensuring that the benchmark reflects what a genuine newcomer might need to know rather than what the system happened to capture.

Questions are distributed equally across five categories, four per category per space. *Direct Visual Search* questions ask where specific objects or equipment are located. *Functional and Description Search* questions ask about the purpose or capabilities of objects and spaces. *Task-Specific Details* questions require step-by-step or procedural knowledge to answer correctly. *Usage and Operational Logic* questions probe practical, situational knowledge a newcomer would need to navigate the space independently. *Edge Cases and Constraints* questions target information that authors are most likely to omit, such as restrictions, maintenance requirements, and failure scenarios. The full question sets for both spaces are provided below.

Lab

Direct Visual Search.

- (1) Where are the AR/VR headsets stored in this lab?
- (2) Where are the shared workstations or compute machines located?
- (3) Where are the monitors located?
- (4) Where do I find cables, adapters, and spare equipment?

Functional and Description Search.

- (1) What is the primary purpose of this space?
- (2) What can I do with the AR/VR headsets in this space?
- (3) What are the robots in this lab used for?
- (4) What kinds of cameras are in this space and what do they capture?

Task-Specific Details.

Manuscript submitted to ACM

- (1) What is the specific process for a new student to use a workstation?
- (2) How do I set up and run an experiment with the AR/VR headsets?
- (3) How do I operate the robots (robot dogs, drones, or toy car)?
- (4) How do I capture a 3D scan or volumetric data of the space?

Usage and Operational Logic.

- (1) Where can I find peripheral devices like monitors, keyboards, and mice to use with the workstations or computers?
- (2) If I am tired, where can I get a coffee or take a break?
- (3) If I buy new cables and connectors for the lab, where and how should I store them?
- (4) I need to run heavy processing software. What can I use?

Edge Cases and Constraints.

- (1) What equipment do I need to charge or maintain after using it?
- (2) Is there anything in this lab I should not touch?
- (3) Who is the primary point of contact if something is needed with the desktops?
- (4) If the wifi goes off, where could I look?

Kitchen

Direct Visual Search.

- (1) Where is the coffee machine located?
- (2) Where is the communal fridge for personal food storage?
- (3) Where can I find the sparkling water machine?
- (4) Where is the toaster?

Functional and Description Search.

- (1) What is the primary purpose of this space?
- (2) I need to heat up my lunch. What object can I use to do this?
- (3) I want to make a cup of tea. Where can I get hot water?
- (4) I need utensils to eat my food. Where can I find them?

Task-Specific Details.

- (1) What is the specific requirement for storing food in the communal fridge to ensure it is not eaten by others?
- (2) Do I have to bring my own coffee or tea?
- (3) Where can I find the supplies needed to make coffee?
- (4) How do I use the coffee machine?

Usage and Operational Logic.

- (1) I want to use the coffee machine; is there a specific training or permission protocol I need to complete first?
- (2) Are there any precautions to use the toaster without accidentally burning my food or causing other problems?
- (3) Are there any precautions or requirements for using the microwave?
- (4) What can I do if I need more details about the happenings and people who use this space?

Edge Cases and Constraints.

- (1) What should I do if the coffee or tea supply runs out in the lab?
- (2) Could anything in the space catch fire? If so, how can I avoid it?
- (3) If my food is missing from the fridge, what could have happened?
- (4) Are there any restrictions on who or when I can access this space?

B Post-Task Questionnaire

The post-task questionnaire was administered after participants completed both the walkthrough and the active elicitation phase. It consisted of four sections: a System Usability Scale (SUS), a NASA Task Load Index (NASA-TLX), a set of custom authoring experience items, and a qualitative reflection.

System Usability Scale

Participants rated the following ten items on a 1–5 scale (Strongly Disagree to Strongly Agree):

- (1) I think that I would like to use this system frequently.
- (2) I found the system unnecessarily complex.
- (3) I thought the system was easy to use.
- (4) I think that I would need the support of a technical person to be able to use this system.
- (5) I found the various functions in this system were well integrated.
- (6) I thought there was too much inconsistency in this system.
- (7) I would imagine that most people would learn to use this system very quickly.
- (8) I found the system very cumbersome to use.
- (9) I felt very confident using the system.
- (10) I needed to learn a lot of things before I could get going with this system.

NASA Task Load Index

Participants rated the following six subscales on a 1–10 scale:

- (1) **Mental Demand:** How mentally demanding was the task?
- (2) **Physical Demand:** How physically demanding was the task?
- (3) **Temporal Demand:** How hurried or rushed was the pace of the task?
- (4) **Performance:** How successful were you in accomplishing what you were asked to do? (1 = Perfect, 10 = Failure)
- (5) **Effort:** How hard did you have to work to accomplish your level of performance?
- (6) **Frustration:** How insecure, discouraged, irritated, stressed, and annoyed were you?

Authoring Experience

Participants rated the following items on a 1–5 scale (Strongly Disagree to Strongly Agree):

- (1) I felt confident that my instructions were complete.
- (2) It was easy to remember to mention important details.
- (3) The system supported explaining spatial relationships effectively.
- (4) I felt that someone unfamiliar with the space could successfully follow my instructions.

Active Elicitation and Interface

Participants rated the following items on a 1–5 scale (Strongly Disagree to Strongly Agree):

- (1) **Detail Recall:** The follow-up questions helped me remember more details.
- (2) **Interruption:** The follow-up questions were frustrating or distracting during the task.
- (3) **Guidance:** The follow-up questions served as a helpful guide for my walkthrough.
- (4) **Instruction Accuracy:** I believe these instructions accurately capture what I demonstrated.
- (5) **Spatial Utility:** The 3D view made it easier to verify if my instructions were correctly placed in the room.

Participants also responded to a binary item: *I found that my instructions were more complete and better after the follow-up questions* (True / False).

Qualitative Reflection

Participants answered three open-ended questions:

- (1) **Helpfulness:** What about the system did you find the most helpful?
- (2) **Improvements:** Is there anything about the system you would change to make it better?
- (3) **Expert Blindness:** Was there anything you realized you forgot only after the system asked you a follow-up?

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009