

ObjChangeVR: Object State Change Reasoning from Continuous Egocentric Views in VR Environments

Shiyi Ding¹, Shaoen Wu¹, Ying Chen²

¹Kennesaw State University, ²Pennsylvania State University
sdging1@students.kennesaw.edu, swu10@kennesaw.edu, yingchen@psu.edu

Abstract

Recent advances in multimodal large language models (MLLMs) offer a promising approach for natural language-based scene change queries in virtual reality (VR). Prior work on applying MLLMs for object state understanding has focused on egocentric videos that capture the camera wearer’s interactions with objects. However, object state changes may occur in the background without direct user interaction, lacking explicit motion cues and making them difficult to detect. Moreover, no benchmark exists for evaluating this challenging scenario. To address these challenges, we introduce ObjChangeVR-Dataset, specifically for benchmarking the question-answering task of object state change. We also propose ObjChangeVR, a framework that combines viewpoint-aware and temporal-based retrieval to identify relevant frames, along with cross-view reasoning that reconciles inconsistent evidence from multiple viewpoints. Extensive experiments demonstrate that ObjChangeVR significantly outperforms baseline approaches across multiple MLLMs.

1 Introduction

Virtual reality (VR) has attracted increasing attention in various fields, such as entertainment, social interactions, and commerce (Fortune Business Insights, 2025). As VR environments become more dynamic and interactive, accurately identifying and localizing scene changes between past and present user views has become an essential task for 3D scene understanding, supporting diverse applications ranging from interactive training simulations (Bjørn et al., 2024) to collaborative virtual workspaces (Jing et al., 2023).

Recent advances in multimodal large language models (MLLMs) have shown remarkable capabilities in scene understanding (OpenAI et al., 2023; Gemini Team et al., 2024; Bai et al., 2023; Liu et al., 2023), offering a promising approach for

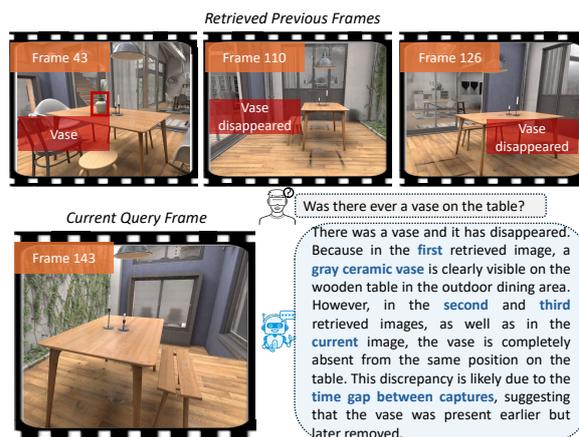


Figure 1: Illustration of the question-answering task for object state change reasoning. Given a query frame and a question about object change, we retrieve several relevant frames from the egocentric frame sequence and leverage visual evidence from the retrieved frames to produce an answer and an explanation.

natural language-based scene change queries in VR environments. Although promising, current scene change detection methods still mainly rely on computer vision techniques that localize pixel-wise change regions (Sachdeva and Zisserman, 2023; Lee and Kim, 2024; Furukawa et al., 2020), lacking support for natural language queries, which allow for a more intuitive and flexible interaction modality for VR users.

Directly applying MLLMs to VR scene change detection faces three challenges. First, users traverse VR environments, generating lengthy frame sequences, but only a small subset contains evidence relevant to a given query on scene change. It is challenging to identify which frames are informative while processing long input sequences. Second, unlike existing egocentric video-based benchmarks (Xiao et al., 2021; Di and Xie, 2024; Ye et al., 2025), object state changes might occur in the background without direct human interaction. For instance, objects may be moved or reconfig-

ured (by another VR user) while a user explores distant areas. Since these background changes lack explicit motion cues and exhibit low perceptual saliency, they are more challenging to detect than changes caused by direct user manipulation captured in egocentric views. Third, no existing benchmark dataset evaluates natural language-based object state change reasoning in continuous egocentric views where user viewpoints shift dramatically across scene sections (e.g., from a kitchen to a study room).

To address these challenges, we introduce a question-answering task on object state change reasoning in continuous egocentric video streams, as shown in Figure 1. We consider a user navigating through an environment, potentially moving across different scene sections and returning to previously visited sections. During this navigation, egocentric frames are continuously captured, and object states may change over time (the process of object state change is not captured in egocentric videos). Given a natural language question such as “*Was there ever a vase on the table?*”, the task involves reasoning over multiple previous frames to determine whether an object ever existed.

We collect a dataset, called ObjChangeVR-Dataset, specifically for benchmarking the object state change question-answering task, and propose a framework, ObjChangeVR, to tackle this task.

First, to identify informative frames from lengthy continuous egocentric frames, ObjChangeVR leverages viewpoint metadata provided by VR devices to retrieve frames containing relevant visual evidence for answering object state change queries. Such viewpoint metadata is increasingly accessible in cameras. Extended reality headsets (e.g., Meta Quest series (Meta, 2026), Apple Vision Pro (Apple, 2026)) automatically record 6-degree-of-freedom (6-DoF) camera pose, and many other consumer devices, such as smartphones (both Android and iOS) or cameras (e.g., GoPro (GoPro, 2026), RealSense (RealSense, Inc., 2026)), can capture visual-inertial and depth sensor data, which can be used to compute camera pose through standard simultaneous localization and mapping (SLAM) pipelines (Cadena et al., 2017), where the pose metadata is then accessible through developer SDK. Therefore, although our current experiments focus on VR, the proposed framework can be applied to real-world egocentric videos when pose information is available or can be reconstructed.

Second, to combat the challenge that object changes occurring without direct user interaction lack explicit motion cues and exhibit low perceptual saliency, ObjChangeVR adopts cross-view reasoning over retrieved frames to improve detection accuracy. Since the retrieved frames are captured from different viewpoints and at different times, they vary in informativeness about the object’s state. ObjChangeVR prioritizes viewpoints with higher informativeness for more reliable reasoning. ObjChangeVR also guides reasoning across temporally ordered frames. For example, when an object consistently appears in earlier frames but is absent in later frames, this pattern provides strong evidence for disappearance rather than mere occlusion. By treating cross-frame inconsistencies as cues rather than noise, our approach effectively distinguishes genuine object state changes from transient observation artifacts.

The main contributions of our paper are: 1) We introduce ObjChangeVR-Dataset, a benchmark for object state change reasoning in continuous egocentric views. The dataset comprises 5 diverse VR scenes (e.g., villa interior, outdoor market) spanning 35 distinct scene sections (e.g., first-floor kitchen, fish shop), with 729 target objects whose states may change over time. 2) We propose ObjChangeVR, which combines viewpoint-aware relevant frame retrieval with a cross-view reasoning module that aggregates and reconciles inconsistent answers from multiple viewpoints to achieve more accurate object change detection. 3) Through experiments, we demonstrate that ObjChangeVR outperforms baseline approaches for frame retrieval and cross-frame inconsistency resolution across multiple MLLMs. Our dataset and code are publicly available at: <https://github.com/sding11/ObjChangeVR>.

2 Related Work

Scene change detection. Scene change detection aims to identify and localize differences between two observations of the same scene captured at different times. Prior studies have leveraged convolutional neural networks (CNNs) to detect changed regions (Noman et al., 2024; Han et al., 2023; Yu et al., 2024; Dong et al., 2025). However, they assume nearly perfect image alignment between the compared frames. While some recent works have addressed unaligned image pairs (Sachdeva and Zisserman, 2023; Lee and Kim, 2024; Furukawa

et al., 2020), they still focus on static image pairs captured from similar viewpoints and do not leverage viewpoint metadata from the camera. In contrast, our task involves analyzing sequences of egocentric frames that exhibit frequent and substantial viewpoint shifts. We incorporate explicit viewpoint metadata available from VR devices to reason about spatial relationships across frames. Moreover, rather than delineating change areas through traditional computer vision techniques, we frame the problem as visual question answering to enable more natural interaction for VR users, offering a new angle on scene change detection that remains largely unexplored.

3D and video-based question answering. Prior work (Wu et al., 2024; Yan et al., 2023; Ma et al., 2023; Yan et al., 2023) on 3D question answering has leveraged 3D scans from datasets like ScanNet (Dai et al., 2017) and focused on processing point cloud data from 3D scenes to respond to specific textual queries about the scenes. A few works have explored using MLLMs for 3D VR scene understanding via situated 3D question answering (Wan et al., 2024; Wu et al., 2023; Ding and Chen, 2025; Li et al., 2025). However, these methods mainly focus on interactive 3D scenes where object changes (if any) are caused by users’ own direct interactions, rather than object changes that occur in the background. Apart from 3D question answering, another line of work addresses video-based question answering (Mogrovejo and Solorio, 2024; Pan et al., 2023; Song et al., 2025). A notable subset of this research focuses on natural language queries in egocentric videos (Xiao et al., 2021; Di and Xie, 2024; Ye et al., 2025), examining human activities to evaluate models’ ability to interpret complex actions and interactions. Complementary to these works, we address object state changes that occur in the background without direct human interaction and without being captured in video recordings. Since these changes lack explicit motion cues and exhibit low perceptual saliency, they present a more challenging question-answering task.

3D and video-based benchmarks for scene understanding. 3D scene datasets such as ScanQA (Azuma et al., 2022), MMScan (Lyu et al., 2024), VLA-3D (Zhang et al., 2024), and SIMMC-VR (Wu et al., 2023) mainly feature static 3D scans without temporal dynamics. Perhaps closest to our work is ChangeSim (Park et al., 2021), which is a benchmark for object state change in a virtual environment. While sequences are cap-

VR scene	Scale	# scene sections	# target objects
Villa interior	Small	9	421
Restaurant	Small	7	62
Market	Small	5	52
Museum	Large	6	134
Viking village	Large	8	60

Table 1: Statistics of ObjChangeVR-Dataset.

tured, ChangeSim operates on paired frames at two discrete timestamps and focuses on annotating pixel-level change detection. In contrast, our collected dataset focuses on natural language queries over extended trajectories. Video-based QA benchmarks such as Causal-VidQA (Li et al., 2022), FunQA (Xie et al., 2024), SurveillanceVQA-589K (Liu et al., 2025), Pano-AVQA (Yun et al., 2021), and VideoEspresso (Han et al., 2025) focus on real-world clips that evaluate event understanding, where the majority of events last for 5 – 20s. Egocentric datasets such as Ego4D (Grauman et al., 2022), EPIC-KITCHENS (Damen et al., 2022), and EgoTracks (Tang et al., 2023) capture human actions, interactions, and object manipulations from a first-person perspective. However, these datasets typically focus on confined scene sections (e.g., a kitchen), where object changes primarily result from direct human interaction. In contrast, ObjChangeVR-Dataset contains trajectories traversing multiple scene sections with drastic viewpoint changes. Additionally, object state changes occur without direct human interaction, meaning that objects may be altered outside the user’s immediate view or control. We summarize the differences of our dataset and existing datasets in Table 8 in Appendix A.2.

3 ObjChangeVR-Dataset

Scene statistics. ObjChangeVR-Dataset uses five distinct VR scenes from the Unity Asset Store, including 4 indoor (Villa interior (ArchVizPRO, 2024), Restaurant (Brick Project Studio, 2023), Market (VallinaArt, 2024), Museum (Leartes Studios, 2024)), and 1 outdoor (Viking village (Unity Technologies, 2022)) scenes. We navigate through each VR scene, mimicking natural exploration in VR. The trajectory is recorded at a frequency of 5Hz, where the camera’s position and orientation are captured continuously over time. Egocentric images are saved at 1Hz. As shown in Table 1, there are 35 scene sections in total (e.g., first-floor kitchen, fish shop, blacksmith workshop). Please see Appendix A.1 for details.

Question statistics. Our dataset is divided into

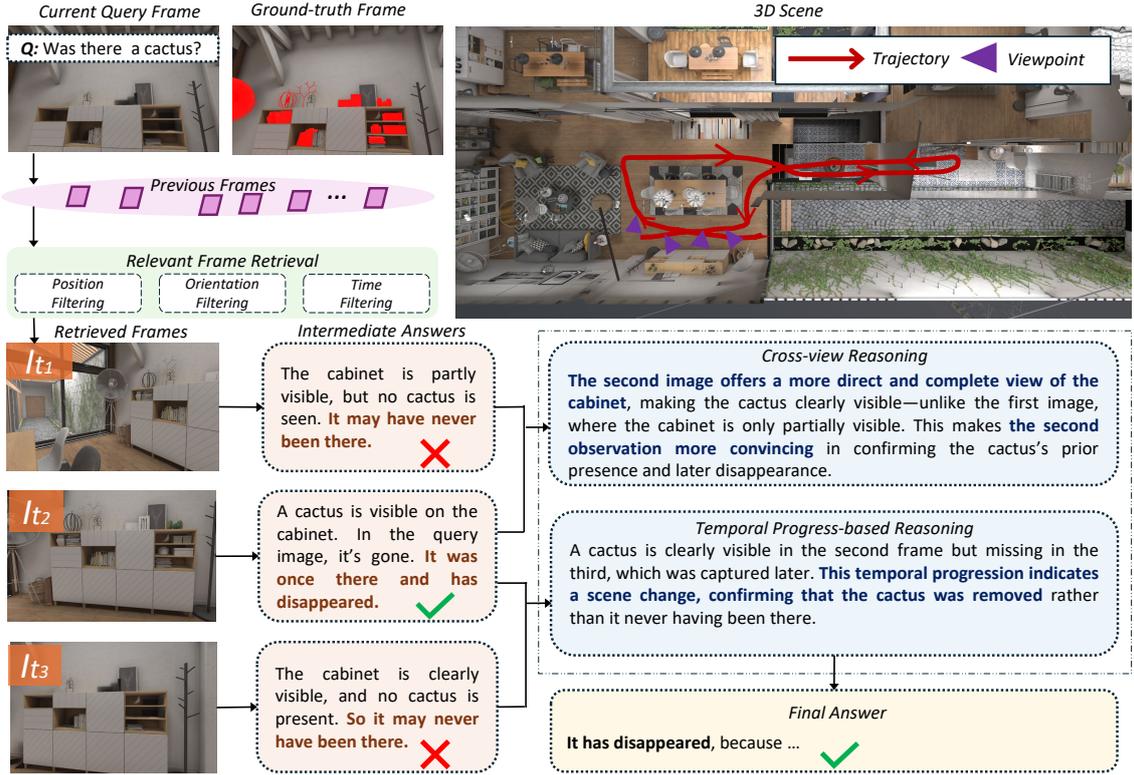


Figure 2: Overview of the ObjChangeVR-Dataset and the proposed ObjChangeVR framework.

two categories based on trajectory length. The *short-trajectory* data comprises approximately 60s walks in three smaller scenes, villa interior, restaurant, and market, with a total of 3,000 questions related to object state change. The *long-trajectory* data comprises approximately 180s walks in two larger scenes, museum and Viking village, with a total of 2,000 questions. Long trajectories cover larger scene scales and temporal variations than short trajectories.

Statistics of target objects. Our question-answering task requires determining whether target objects have changed their states by the time their scene section is revisited and they are observed from different perspectives. The target objects vary in scale and complexity, including large objects (e.g., dinosaur skeletons, barrel stacks), small items (e.g., vases, photo frames, cookies), and grouped objects that disappear together (e.g., tables with their associated items). Across all scenes, there are 729 target objects in total, as shown in Table 1.

User trajectories. Data is recorded automatically as we traverse the virtual environments (Figure 2 shows an example of a 3D scene and a trajectory), capturing target objects scattered across the scene through an egocentric view. As the trajectory continues, previously seen objects naturally leave the

camera’s field of view as we explore new scene sections. At a specific point during the walk, certain objects change their state (e.g., disappear) while others remain unchanged. Users then revisit the same scene sections, but without retracing their exact steps. Instead, they return along different routes with varied angles and distances to the target objects. This means that the objects are observed from new perspectives, for example, passing an area from the opposite direction or at a different distance. Although these viewpoints still include the areas where the objects were once observed, the positional and angular differences of viewpoints make it difficult to align observations across time. This setup creates a realistic temporal reasoning challenge, allowing us to evaluate whether models can determine object state changes over time while handling viewpoint variation between current frame and previous frames.

Annotations. We develop a semi-automated pipeline for ground-truth annotations, which leverages both the Unity game engine (which generates object-level masks) and the reasoning capabilities of MLLMs, followed by human verification. We use Unity to overlay object masks directly onto ground-truth frames (the ground-truth frame is shown in Figure 2), highlighting objects that

have changed their state in a high-contrast color (e.g., red). These annotated images, along with explicit instructions that the red-highlighted regions correspond to objects that were previously present but have now disappeared, are then provided to an MLLM (GPT-4o) to generate ground-truth answers to questions about objects’ state change. Finally, human reviewers verify and correct the generated annotations to ensure correctness.

4 Methodology

Consider a user navigating through an environment, potentially moving across different scene sections and returning to previously visited sections. During this navigation, egocentric frames are continuously captured, and object states may change over time. Let $H_t = \{I_i \mid i < t\}$ denote the set of all frames captured before the current timestamp t , where I_i is the frame recorded at time i . Let I_c denote the current egocentric frame. Each frame is associated with the camera’s viewpoint location (x , y , and z coordinates), orientation (represented as a quaternion (q_w, q_x, q_y, q_z)), and timestamp. Given a current frame I_c and a natural language question Q about the state of a specific object (e.g., whether the object has disappeared, has always been present, or was never there), the goal is to generate a natural language answer and explanation indicating the object’s state change. To support this temporal reasoning task, we retrieve a set of previous egocentric frames from the past frame set H_t (Section 4.1). Specifically, we select up to k past frames $\{I_{t_j}\}_{j=1}^k \subseteq H_t$ that are most relevant to the question Q , i.e., most likely to contain information that helps determine the object’s state, where timestamps $t_1 < t_2 < \dots < t_k < t$ indicate the chronological order in which these past frames were captured. These selected frames, along with I_c and Q , are provided to the MLLM as additional context to generate the final answer (Section 4.2). k is a configurable hyperparameter, which is chosen based on computational cost or limits on MLLM input size (e.g., token budget).

4.1 Relevant Cross-view Frame Retrieval

A natural approach to retrieving relevant frames is to compare the visual embeddings of past frames with the current query frame I_c . However, this may return misleading matches that look similar but come from different areas. For instance, hallways may have the same wall colors, flooring, and

chairs arranged in similar ways. To address this, we leverage sensor data automatically recorded by VR devices, including the camera position and orientation associated with each frame. This ensures that retrieved frames not only share visual similarity but also originate from the correct spatial region.

To retrieve relevant past frames, we apply a three-stage hierarchical filtering: position filtering, orientation filtering, and temporal filtering. First, *position filtering* selects the top k_p previous frames whose viewpoint positions have the smallest Euclidean distance to the position \mathbf{p}_c of the current frame I_c , ensuring spatial proximity. Among these frames, *orientation filtering* further keeps the top k_o frames whose viewpoint orientations are most closely aligned with the orientation (represented as a quaternion \mathbf{q}_c) of the current frame I_c , favoring similar viewing angles. Finally, *temporal filtering* selects the earliest k frames from the orientation-filtered set to maintain chronological diversity.

Next, we dynamically adjust cutoff values k_p and k_o for the position-filtered and orientation-filtered sets based on how many relevant frames the system ultimately needs to select. If k increases but k_p and k_o remain small, relevant frames may be prematurely discarded, resulting in low retrieval recall. Conversely, if k_p and k_o are too large, the intermediate filters may admit too many redundant candidates, diminishing the effectiveness of the final filter. To balance retrieval precision and recall, we grow the position-filtered and orientation-filtered sets proportionally to k , while bounding their sizes within configurable minimum and maximum limits to avoid overly narrow or large selections:

$$k_o = \min\left(|H_t|, \text{cap}_o, \max\left(\min_o, \lceil \alpha k \rceil\right)\right),$$

$$k_p = \min\left(|H_t|, \text{cap}_p, \max\left(\min_p, \lceil \beta k_o \rceil\right)\right),$$

where α and β are linear scaling factors that control the size of the orientation- and position-filtered sets relative to the final budget k , \min_o and \min_p ensure the filters are not overly restrictive when k is small, and cap_o and cap_p limit the set sizes to avoid excessive growth when k is large.

4.2 Temporal Cross-view Reasoning

After retrieving k previous frames $\{I_{t_j}\}_{j=1}^k$ that are likely to provide relevant context for answering an object state change question, we aim to let the MLLM output the final answer about the object state change. Our strategy incorporates temporal

cross-view information and exploits a two-stage chain-of-thought (CoT) prompting that 1) performs pairwise reasoning between each retrieved frame and the current frame to generate k intermediate answers, and 2) derives the final answer by aggregating and reconciling these intermediate answers when inconsistencies arise.

Independent intermediate answers. We provide few-shot exemplars that guide the MLLM through comparing each retrieved frame and the current frame, describing salient visual differences, and answering whether the object has changed. This process yields independent k intermediate answers, each accompanied by a brief explanation that combines information from both the current frame and the corresponding retrieved frame to justify the determination of object state change.

The final answer from temporal cross-view reasoning. We then derive the final answer from the intermediate k answers. When k intermediate answers are consistent, ObjChangeVR adopts the consensus. When intermediate answers are inconsistent, ObjChangeVR reconciles them by jointly accounting for viewpoint variations and leveraging the chronological order of the frames. This enables the prioritization of the most informative observations while reasoning about temporal progression.

(1) *Cross-view reasoning.* Since the retrieved frames are captured from different viewpoints and at different times, they vary in how informative they are about the object’s state. Factors such as occlusion or camera angle may cause some frames to miss the object even when it is present, leading to misleading conclusions. To address this, we include inputs to the MLLM that explicitly guide the model in reconciling such inconsistencies through cross-validation across multiple frames. We also add few-shot exemplars. When intermediate answers disagree with each other, the model evaluates which frames offer more reliable visual cues. For example, an exemplar instructs that if most frames show no target object but one frame clearly captures it, it can be inferred that the missing object in other frames may be due to poor viewing angles or occlusions, rather than the conclusion that the object never existed. By recognizing and attributing discrepancies to uninformative viewpoints, the model is better equipped to detect true state changes.

(2) *Temporal progress-based reasoning.* An object may appear in earlier frames and change its state in later ones, meaning that temporal changes

themselves can lead to inconsistent intermediate answers. Suppose the retrieved frames $\{I_{t_j}\}_{j=1}^k$ are captured at ordered timestamps (t_1, t_2, \dots, t_k) . We feed these frames to the MLLM in their chronological order, allowing it to reason with respect to temporal progression. We include instructions (including few-shot exemplars) that explicitly demonstrate how to analyze object presence across ordered frames. For instance, when an object exhibits a consistent presence in earlier timestamps but then disappears in later ones, this temporal trajectory provides strong evidence for an actual disappearance event. Rather than treating such inconsistencies as noise, our approach leverages them to identify genuine state changes and infer that the object has disappeared.

5 Experimental Setup

5.1 Baselines

To evaluate the effectiveness of retrieving relevant egocentric frames, we compare ObjChangeVR with Caption-CLIP, Image-CLIP, and Viewpoint-Retrieval. To examine the impact of reasoning strategies, we further compare ObjChangeVR against CoT-SC (Wang et al., 2022) adapted to our task, as well as ObjChangeVR without the temporal cross-viewpoint CoT, i.e., ObjChangeVR w/o TCv.

Caption-CLIP. Each frame is captioned using an MLLM (GPT-4o). Similar to the relevant frame retrieval method used in Tang et al. (2024), we encode the question and the captions of all previous frames using CLIP (Radford et al., 2021), and rank the captions by their cosine similarity to the question in the embedding space. The MLLM is then prompted with textual input consisting of the question, the caption of the current frame, and the top- k most relevant captions from previous frames.

Image-CLIP. Similar to the relevant frame retrieval method in Liang and Albanie (2023), we embed the current frame and all previous frames with CLIP and rank the previous frames by their cosine similarity to the current frame in the visual embedding space. The MLLM is then prompted with multimodal input consisting of the question, the current frame, and the k most similar retrieved frames, as in our ObjChangeVR.

Viewpoint-Retrieval. Previous frames are ranked based solely on the difference between their camera viewpoints and the current camera viewpoint. Formally, the similarity score for a previous frame

i with the current frame is defined as $\text{sim}(i) = w_p \cdot d_{\text{pos}}(i) + w_o \cdot d_{\text{orn}}(i)$, where $d_{\text{pos}}(i)$ denotes the Euclidean distance between the frame i 's viewpoint position and the current viewpoint position, and $d_{\text{orn}}(i)$ represents the angular difference between their orientations. We empirically set $w_p = w_o = 1$. As in our ObjChangeVR, the MLLM is then prompted with multimodal input.

CoT-SC (Wang et al., 2022). Adapting CoT-SC to our task, we conduct S independent inference runs on the same inputs: the question, current image, and the k retrieved historical images. The retrieval pipeline is the same as that of ObjChangeVR. We generate outputs using a softmax temperature of $t = 0.7$ and shuffle the order of retrieved frames to encourage diversity. The final answer is obtained by aggregating the S results via majority vote. For fair comparison, we set $S = 3$ to match the three reasoning steps used in ObjChangeVR.

5.2 Evaluation Metrics

Exact match (EM) $@\tau$. We extract the judgment clause (e.g., the clause that states whether the object disappeared) from each answer and compute a similarity score against the ground truth using normalized string matching. Strict EM, the proportion of predictions that exactly match the ground-truth answers at the string level, remains close to zero in our task. This is because the generated answers include explanatory text beyond the concise ground-truth labels (e.g., disappeared, always here). To address this, we count a prediction as correct if its similarity to the ground truth exceeds a threshold τ . This relaxed metric, denoted as EM $@\tau$, captures near-exact correctness while tolerating wording differences. We set $\tau = 0.80$, informed by the sensitivity analysis showing that our results are insensitive to the choice of τ in $\{0.70, 0.75, 0.80, 0.85, 0.90\}$ (Table 9 in Appendix A.3).

Macro-F1. To evaluate categorical accuracy, both the generated answer and the ground-truth answer are classified into one of three predefined semantic classes (disappeared, never there, or always been there) based on indicative phrases or keywords in their respective judgment clauses. We compare the generated and the ground-truth answers using EM $@\tau$. The macro-F1 score is computed by calculating the F1 score for each class independently and averaging them equally.

Weighted-F1. We report weighted-F1, which averages the per-class F1 scores weighted by the num-

ber of samples per class. It reflects overall performance under the imbalanced class distribution.

5.3 Hyperparameters and Default Settings

By default, we set $k = 3$. For retrieving relevant past frames in Section 4.1, we set $\alpha = 2$ and $\beta = 2$, $(\min_o, \text{cap}_o) = (7, 30)$, and $(\min_p, \text{cap}_p) = (30, 80)$. GPT-4o is used as the default MLLM.

6 Experimental Results

Overall performance of ObjChangeVR. Tables 2 and 3 report EM $@0.8$, macro-F1, and weighted-F1 scores on both short and long video trajectories. Table 2 shows the comparison of ObjChangeVR against different relevant frame retrieval methods, while Table 3 presents the comparison of ObjChangeVR against variants with different reasoning methods. For the question-answering task on object state change, ObjChangeVR consistently outperforms all other methods across both types of video recordings on all metrics. In particular, using GPT-4o as the MLLM, ObjChangeVR achieves an EM $@0.8$ of 0.822 on short trajectories and 0.652 on long trajectories, resulting in an overall average EM $@0.8$ of 0.754. Interestingly, Viewpoint-Retrieval outperforms both Caption-CLIP and Image-CLIP across short and long video recordings, indicating that the position and orientation of viewpoints prove valuable for retrieving relevant frames in VR environments. Compared with Caption-CLIP, Image-CLIP, and Viewpoint-Retrieval, the superior performance of ObjChangeVR suggests that it enables more effective retrieval of frames containing information to answer questions regarding object state change. The performance gains over CoT-SC and ObjChangeVR w/o TCV confirm the effectiveness of reasoning about the retrieved frames, reconciling conflicting cues across frames, and producing more accurate answers.

Impact of MLLMs. Tables 2 and 3 also show that ObjChangeVR consistently outperforms all other methods with different MLLMs (GPT-4o, GPT-4o mini, and Gemini 2.0 Flash), demonstrating its effectiveness regardless of model size or architecture. The performance gains over other methods vary across models: for instance, with GPT-4o, ObjChangeVR outperforms CoT-SC by 5.9% in EM $@0.8$, while with GPT-4o mini and Gemini 2.0 Flash, the improvements are 11.7% and 8.4%, respectively. Smaller models benefit more from our

Model	Method	Short traj.			Long traj.			All traj.		
		EM @0.8	Macro F1	Weighted F1	EM @0.8	Macro F1	Weighted F1	EM @0.8	Macro F1	Weighted F1
GPT-4o	Caption-CLIP	0.529	0.595	0.588	0.528	0.522	0.537	0.529	0.574	0.570
	Image-CLIP	0.616	0.649	0.644	0.558	0.572	0.585	0.592	0.619	0.620
	Viewpoint-Retrieval	0.623	0.659	0.657	0.570	0.584	0.599	0.601	0.631	0.635
	ObjChangeVR	0.822	0.830	0.837	0.652	0.661	0.669	0.754	0.770	0.774
GPT-4o mini	Caption-CLIP	0.416	0.395	0.357	0.537	0.449	0.468	0.464	0.420	0.400
	Image-CLIP	0.472	0.430	0.381	0.513	0.459	0.479	0.489	0.441	0.420
	Viewpoint-Retrieval	0.472	0.435	0.388	0.527	0.471	0.491	0.494	0.449	0.429
	ObjChangeVR	0.696	0.692	0.699	0.589	0.573	0.579	0.653	0.656	0.657
Gemini 2.0 Flash	Caption-CLIP	0.478	0.493	0.453	0.440	0.430	0.453	0.462	0.471	0.455
	Image-CLIP	0.643	0.667	0.661	0.563	0.571	0.582	0.611	0.630	0.630
	Viewpoint-Retrieval	0.653	0.677	0.672	0.590	0.594	0.605	0.628	0.645	0.645
	ObjChangeVR	0.786	0.806	0.811	0.604	0.615	0.624	0.713	0.739	0.741

Table 2: Comparison of ObjChangeVR and different relevant frame retrieval methods across three MLLMs.

Model	Method	Short traj.			Long traj.			All traj.		
		EM @0.8	Macro F1	Weighted F1	EM @0.8	Macro F1	Weighted F1	EM @0.8	Macro F1	Weighted F1
GPT-4o	CoT-SC	0.754	0.745	0.754	0.607	0.623	0.631	0.695	0.702	0.708
	ObjChangeVR w/o TCV	0.745	0.737	0.747	0.590	0.611	0.620	0.683	0.699	0.700
	ObjChangeVR	0.822	0.830	0.837	0.652	0.661	0.669	0.754	0.770	0.774
GPT-4o mini	CoT-SC	0.537	0.536	0.518	0.524	0.536	0.549	0.536	0.545	0.532
	ObjChangeVR w/o TCV	0.527	0.528	0.511	0.536	0.536	0.549	0.531	0.541	0.529
	ObjChangeVR	0.696	0.692	0.699	0.589	0.573	0.579	0.653	0.656	0.657
Gemini 2.0 Flash	CoT-SC	0.688	0.703	0.709	0.541	0.559	0.565	0.629	0.649	0.653
	ObjChangeVR w/o TCV	0.669	0.682	0.689	0.533	0.556	0.562	0.615	0.636	0.639
	ObjChangeVR	0.786	0.806	0.811	0.604	0.615	0.624	0.713	0.739	0.741

Table 3: Comparison of ObjChangeVR and variants with different reasoning methods across three MLLMs.

Metric	Case	CoT-SC	ObjChangeVR
EM@0.8	Cons.	0.748	0.795
	Incons.	0.637	0.709
	All	0.695	0.754
Macro-F1	Cons.	0.520	0.749
	Incons.	0.551	0.624
	All	0.702	0.770
Weighted-F1	Cons.	0.700	0.810
	Incons.	0.664	0.720
	All	0.708	0.774

Table 4: Performance of ObjChangeVR under consistent and inconsistent intermediate answers.

approach, suggesting that ObjChangeVR’s retrieval and reasoning framework helps compensate for the performance gap in smaller-scale models.

Performance under consistent and inconsistent intermediate answers. We evaluate the reasoning capability under consistent and inconsistent intermediate answers. We test both CoT-SC and ObjChangeVR and report results in Table 4, as these

Metric	Traj.	k						
		1	2	3	5	7	9	
EM @0.8	Short	0.691	0.774	0.822	0.776	0.711	0.672	
	Long	0.542	0.638	0.652	0.598	0.554	0.532	
	All	0.631	0.720	0.754	0.705	0.648	0.616	
Macro-F1	Short	0.787	0.832	0.830	0.823	0.812	0.801	
	Long	0.612	0.656	0.661	0.656	0.648	0.651	
	All	0.727	0.770	0.770	0.762	0.751	0.743	
Weighted-F1	Short	0.794	0.838	0.837	0.831	0.824	0.812	
	Long	0.621	0.664	0.669	0.662	0.654	0.655	
	All	0.730	0.773	0.774	0.767	0.758	0.750	

Table 5: Performance of ObjChangeVR across different numbers of retrieved frames (k).

are two methods that use intermediate answers. CoT-SC produces inconsistent intermediate reasoning answers in 47.9% of the questions. In comparison, ObjChangeVR reduces this inconsistency ratio to 33.2%, demonstrating an improvement in producing consistent reasoning paths. Across all metrics, ObjChangeVR outperforms CoT-SC un-

Variant	Zero-shot	Few-shot
EM	0.731	0.754
Macro-F1	0.734	0.770
Weighted-F1	0.740	0.774

Table 6: Comparison between the ObjChangeVR variants with zero-shot and few-shot prompting.

der both consistent and inconsistent intermediate answers. In particular, under inconsistent intermediate answers, ObjChangeVR achieves 7.2%, 7.3%, and 5.6% improvements in EM@0.8, macro-F1, and weighted-F1 over CoT-SC, respectively. These results demonstrate that ObjChangeVR’s temporal cross-view reasoning enhances robustness by generating reliable final answers even with varying reasoning paths.

Impact of k . We examine the impact of the number of retrieved frames (k) in Table 5. For different k , ObjChangeVR consistently performs better on short trajectories than on long ones, suggesting that it is more effective when the temporal spans and viewpoint changes are smaller along the trajectory. As k increases from 1 to 3, we observe consistent improvements in EM@0.8, macro-F1, and weighted-F1 scores. This indicates that retrieving multiple frames provides richer contextual information compared to relying on a single frame, where additional frames capture objects from different viewpoints or time periods.

As k increases from 3 to 9, we observe a performance decline. For instance, EM@0.8 decreases by 15.0% on short trajectories and 12.0% on long ones. To investigate the reason behind this trend, we analyze how intermediate reasoning inconsistency affects final answers. Figure 3 in Appendix A.4 shows a drop in the proportion of consistent intermediate answers (with a 31.0% drop from $k = 3$ to $k = 9$), indicating that retrieving more frames increases the chance of introducing conflicting or distracting contextual information as k grows too large. This rise in inconsistent reasoning contributes to the performance decline observed at larger k . Based on the observation, we select $k = 3$ as the default setting in the paper. Retrieving a small number of previous frames (e.g., 3) achieves the best performance by providing richer contextual information than a single frame, while reducing the negative impacts brought by inconsistent intermediate reasoning for larger k . It also reduces token consumption and inference latency.

Impact of τ . We examine whether our evaluation results are sensitive to the choice of the

similarity threshold τ used in EM@ τ . Specifically, we evaluate all methods with $\tau \in \{0.70, 0.75, 0.80, 0.85, 0.90\}$. As shown in Table 9 in Appendix A.3, the relative ranking of all methods remains unchanged across different τ , and ObjChangeVR consistently achieves the best performance. Moreover, the absolute variations in EM@ τ across different τ are small, indicating that our conclusions are not sensitive to the specific choice of τ .

Impact of few-shot prompting. To better understand the role of few-shot prompting in ObjChangeVR, we evaluate a zero-shot variant. In this variant, we remove all few-shot exemplars from both the pairwise comparison prompt and the final reasoning prompt. The prompts only describe how to reason (e.g., using the temporal ordering of retrieved frames and the spatial similarity of viewpoints), without providing any exemplars. Table 6 reports the comparison between the ObjChangeVR variants with zero-shot and few-shot prompting. It shows that removing all exemplars leads to only a small performance decrease, and ObjChangeVR with zero-shot prompting still outperforms all other methods (the performance of other methods is listed in Tables 2 and 3). These findings suggest that while few-shot prompting provides a modest benefit, the effectiveness of our reasoning module does not heavily rely on it.

7 Conclusion

In this paper, we introduce the ObjChangeVR-Dataset for the question-answering task of object change detection with continuous egocentric views. We also propose ObjChangeVR to effectively retrieve relevant frames that contain useful information to answer the object state change query, and then use cross-view reasoning and temporal progress-based reasoning to get a final answer. Experimental results demonstrate that ObjChangeVR achieves higher reasoning accuracy compared with various methods and variants across both short and long trajectories and under consistent and inconsistent intermediate reasoning answers.

Acknowledgment

We thank the anonymous reviewers for their constructive comments. This work was supported in part by NSF grant No. 2550742.

Limitations

Our study has several limitations. First, due to limited computational resources, we were unable to deploy MLLMs capable of processing multiple retrieved images in a single prompt on local servers or workstations. Hence, we could not evaluate several representative MLLMs locally. Second, we primarily focus on object state changes where an object disappears. We focus on object disappearance as it represents a particularly challenging scenario: it requires reasoning about what may be no longer visible in the current query image. Other change types (e.g., object additions or movements) are yet to be explored. To provide preliminary insights, we collected a small-scale dataset on object additions and evaluated our method on it in Appendix A.5. Third, our data collection process requires manual trajectory sampling and partial human verification, limiting our ability to scale the dataset.

Ethical Considerations

All data collection in this study was conducted entirely within virtual environments. User trajectories were generated through manual navigation in VR, thereby eliminating the risk of human injury. This study does not involve human subjects, and the collected data does not involve personal or sensitive information. All VR scenes used for data generation were obtained through purchase and are permitted for academic research. Furthermore, our use of MLLMs adheres to applicable laws, licensing terms, and institutional guidelines.

References

- Apple. 2026. [Apple Vision Pro](#).
- ArchVizPRO. 2024. [ArchVizPRO interior vol.6](#).
- Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. 2022. ScanQA: 3D question answering for spatial scene understanding. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.
- Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, and 1 others. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.
- Pernille Bjørn, Maja Ling Han, Andrea Parezanovic, and Per Larsen. 2024. Social fidelity in cooperative virtual reality maritime training. *Human-Computer Interaction*.
- Brick Project Studio. 2023. [Fast food restaurant kit](#).
- Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian Reid, and John J Leonard. 2017. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on robotics*, 32(6):1309–1332.
- Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. 2017. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In *Proc. IEEE CVPR*.
- Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Jian Ma, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. 2022. Rescaling egocentric vision: Collection, pipeline and challenges for EPIC-KITCHENS-100. *International Journal of Computer Vision (IJCV)*.
- Shangzhe Di and Weidi Xie. 2024. Grounded question-answering in long egocentric videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- Shiyi Ding and Ying Chen. 2025. RAG-VR: Leveraging retrieval-augmented generation for 3D question answering in VR environments. In *2025 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. IEEE.
- Sijun Dong, Fangcheng Zuo, Geng Chen, Siming Fu, and Xiaoliang Meng. 2025. A remote sensing image change detection method integrating layer-exchange and channel-spatial differences. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*.
- Fortune Business Insights. 2025. [Virtual reality \(VR\) market size, share & industry analysis, 2025–2032](#).
- Yukuko Furukawa, Kumiko Suzuki, Ryuhei Hamaguchi, Masaki Onishi, and Ken Sakurada. 2020. Self-supervised simultaneous alignment and change detection. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6025–6031. IEEE.
- Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, and 1 others. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.
- GoPro. 2026. [The original action camera](#).
- Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh Kumar Ramakrishnan, Fiona Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Zhongcong Xu, and 66 others. 2022. Ego4D: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

- Chengxi Han, Chen Wu, Haonan Guo, Meiqi Hu, and Hongruixuan Chen. 2023. HANet: A hierarchical attention network for change detection with bitemporal very-high-resolution remote sensing images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*.
- Songhao Han, Wei Huang, Hairong Shi, Le Zhuo, Xiu Su, Shifeng Zhang, Xu Zhou, Xiaojuan Qi, Yue Liao, and Si Liu. 2025. VideoEspresso: A large-scale chain-of-thought dataset for fine-grained video reasoning via core frame selection. In *Proceedings of the Computer Vision and Pattern Recognition Conference*.
- Allison Jing, Michael Frederick, Monica Sewell, Amy Karlson, Brian Simpson, and Missie Smith. 2023. How visualising emotions affects interpersonal trust and task collaboration in a shared virtual space. In *2023 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE.
- Leartes Studios. 2024. [Historical museum](#).
- Seonhoon Lee and Jong-Hwan Kim. 2024. Semi-supervised scene change detection by distillation from feature-metric alignment. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1226–1235.
- Jiangtong Li, Li Niu, and Liqing Zhang. 2022. From representation to reasoning: Towards both evidence and commonsense reasoning for video question-answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.
- Ziming Li, Huadong Zhang, Chao Peng, and Roshan Peiris. 2025. Exploring large language model-driven agents for environment-aware spatial interactions and conversations in virtual reality role-play scenarios. In *2025 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*. IEEE.
- Kaiqu Liang and Samuel Albanie. 2023. Simple baselines for interactive video retrieval with questions and answers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*.
- Bo Liu, Pengfei Qiao, Minhan Ma, Xuange Zhang, Yinan Tang, Peng Xu, Kun Liu, and Tongtong Yuan. 2025. SurveillanceVQA-589K: A benchmark for comprehensive surveillance video-language understanding with large models. *arXiv preprint arXiv:2505.12589*.
- Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Advances in neural information processing systems*.
- Ruiyuan Lyu, Jingli Lin, Tai Wang, Shuai Yang, Xiaohan Mao, Yilun Chen, Runsen Xu, Haifeng Huang, Chenming Zhu, Dahua Lin, and 1 others. 2024. MM-Scan: A multi-modal 3D scene dataset with hierarchical grounded language annotations. *Advances in Neural Information Processing Systems*.
- Xiaojuan Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. 2023. SQA3D: Situated question answering in 3D scenes. In *Proc. IEEE ICLR*.
- Meta. 2026. [Meta Quest: VR headsets and accessories](#).
- David Mogrovejo and Thamar Solorio. 2024. Question-instructed visual descriptions for zero-shot video answering. In *Findings of the Association for Computational Linguistics ACL 2024*.
- Mubashir Noman, Mustansar Fiaz, Hisham Cholakkal, Salman Khan, and Fahad Shahbaz Khan. 2024. ELGC-Net: Efficient local–global context aggregation for remote sensing change detection. *IEEE Transactions on Geoscience and Remote Sensing*.
- OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mo Bavarian, Jeff Belgum, and 261 others. 2023. [GPT-4 technical report](#).
- Junting Pan, Ziyi Lin, Yuying Ge, Xiatian Zhu, Renrui Zhang, Yi Wang, Yu Qiao, and Hongsheng Li. 2023. Retrieving-to-answer: Zero-shot video question answering with frozen large language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.
- Jin-Man Park, Jae-Hyuk Jang, Sahng-Min Yoo, Sun-Kyung Lee, Ue-Hwan Kim, and Jong-Hwan Kim. 2021. ChangeSim: Towards end-to-end online scene change detection in industrial indoor environments. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE.
- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. PmlR.
- RealSense, Inc. 2026. [RealSense](#).
- Ragav Sachdeva and Andrew Zisserman. 2023. The change you want to see. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3993–4002.
- Enxin Song, Wenhao Chai, Tian Ye, Jenq-Neng Hwang, Xi Li, and Gaoang Wang. 2025. MovieChat+: Question-aware sparse memory for long video question answering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Chia-Wei Tang, Ting-Chih Chen, Kiet A. Nguyen, Kazi Sajeed Mehrab, Alvi Md Ishmam, and Chris Thomas. 2024. M3D: MultiModal MultiDocument fine-grained inconsistency detection. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

- Hao Tang, Kevin J Liang, Kristen Grauman, Matt Feiszli, and Weiyao Wang. 2023. EgoTracks: A long-term egocentric visual object tracking dataset. *Advances in Neural Information Processing Systems*, 36.
- Unity Technologies. 2022. [Viking village URP](#).
- VallinaArt. 2024. [Low-poly Medieval market](#).
- Hongyu Wan, Jinda Zhang, Abdulaziz Arif Suria, Bingsheng Yao, Dakuo Wang, Yvonne Coady, and Mirjana Prpa. 2024. Building LLM-based AI agents in social virtual reality. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*.
- Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.
- Te-Lin Wu, Satwik Kottur, Andrea Madotto, Mahmoud Azab, Pedro Rodriguez, Babak Damavandi, Nanyun Peng, and Seungwhan Moon. 2023. SIMMC-VR: A task-oriented multimodal dialog dataset with situated and immersive VR streams. In *Proc. of ACL*.
- Zizhao Wu, Haohan Li, Gongyi Chen, Zhou Yu, Xiaoling Gu, and Yigang Wang. 2024. 3D question answering with scene graph reasoning. In *Proc. ACM MM*.
- Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. 2021. NEX-T-QA: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.
- Binzhu Xie, Sicheng Zhang, Zitang Zhou, Bo Li, Yuanhan Zhang, Jack Hessel, Jingkang Yang, and Ziwei Liu. 2024. FunQA: Towards surprising video comprehension. In *European Conference on Computer Vision*. Springer.
- Xu Yan, Zhihao Yuan, Yuhao Du, Yinghong Liao, Yao Guo, Shuguang Cui, and Zhen Li. 2023. Comprehensive visual question answering on point clouds through compositional scene manipulation. *IEEE Transactions on Visualization and Computer Graphics*, 30(12):7473–7485.
- Hanrong Ye, Haotian Zhang, Erik Daxberger, Lin Chen, Zongyu Lin, Yanghao Li, Bowen Zhang, Haoxuan You, Dan Xu, Zhe Gan, and 1 others. 2025. MM-Ego: Towards building egocentric multimodal LLMs for video QA. In *Proc. ICLR*.
- Weikang Yu, Xiaokang Zhang, Samiran Das, Xiao Xi-ang Zhu, and Pedram Ghamisi. 2024. MaskCD: A remote sensing change detection network based on mask classification. *IEEE Transactions on Geoscience and Remote Sensing*.
- Heeseung Yun, Youngjae Yu, Wonsuk Yang, Kangil Lee, and Gunhee Kim. 2021. Pano-AVQA: Grounded audio-visual question answering on 360deg videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.
- Haochen Zhang, Nader Zantout, Pujith Kachana, Zongyuan Wu, Ji Zhang, and Wenshan Wang. 2024. VLA-3D: A dataset for 3D semantic scene understanding and navigation. *arXiv preprint arXiv:2411.03540*.

A Appendix

A.1 Details of ObjChangeVR-Dataset

Scene scales and sections in ObjChangeVR-Dataset are listed in Table 7.

A.2 Comparison of ObjChangeVR-Dataset and Existing 3D and Video-based Benchmarks

We compare the ObjChangeVR-Dataset with representative 3D and video-based benchmarks across multiple dimensions. The comparison is summarized in Table 8.

A.3 Impact of the Similarity Threshold τ

Table 9 shows the impact of the threshold τ on the performance of different methods.

A.4 Influence of k on the Intermediate Answers

The number of consistent and inconsistent intermediate reasoning answers under different k is shown in Figure 3. As k increases, the proportion of consistent intermediate answers gradually decreases, while the proportion of inconsistent answers rises.

A.5 Object Addition

Our pipeline can potentially be generalized to different change types (e.g., object additions and movements). To demonstrate this potential, we collected a small-scale dataset of 100 questions on object additions using our established pipeline and evaluated our method on this dataset. The results are shown in Table 10. In future work, we plan to expand this benchmark by constructing full-scale datasets for categories such as object movement and addition.

A.6 Inference Latency

We investigate the inference latency by measuring retrieval time, reasoning time, and total time for each method (Table 11). We use GPT-4o as

Scene	Scale	Scene section
Villa interior	Small	First-floor kitchen, first-floor dining room, first-floor living room, first-floor balcony, first-floor lounge, first-floor storage room, second-floor bathroom, second-floor master bedroom with bed and wardrobe, second-floor study room with bookshelf and desk
Restaurant	Small	Dining area, men’s restroom, women’s restroom, entrance, checkout counter, pickup area, bar counter
Market	Small	Fish shop, dessert shop, butcher shop, weapon shop, vegetable stall
Museum	Large	Ground floor main exhibition hall, ground floor small exhibition room, ground floor corridor exhibition area, mezzanine staircase area, second floor corridor exhibition area, second floor balcony
Viking village	Large	Large house, small house, barrel storage, blacksmith workshop, dock, shipbuilding area, gate and tower, fenced area with multiple items

Table 7: Scene sections in ObjChangeVR-Dataset.

Dataset	Virtual env.-based	Natural language Q&A	Egocentric	Object state change	Camera pose	Avg. duration	Scene type
ScanQA (Azuma et al., 2022)	✗	✓	✗	✗	✓	N/A	Indoor
MMScan (Lyu et al., 2024)	✗	✓	✗	✗	✓	N/A	Indoor
VLA-3D (Zhang et al., 2024)	✓	✗	✗	✗	✗	N/A	Indoor
SIMMC-VR (Wu et al., 2023)	✓	✓	✓	✗	✓	~2 min	Indoor
ChangeSim (Park et al., 2021)	✓	✗	✗	✓	✓	N/A	Indoor
Causal-VidQA (Li et al., 2022)	✗	✓	✗	✓	✗	>9 s	Indoor/Outdoor
FunQA (Xie et al., 2024)	✗	✓	✗	✓	✗	19 s	Indoor/Outdoor
SurveillanceVQA (Liu et al., 2025)	✗	✓	✗	✓	✗	N/A	Indoor/Outdoor
Pano-AVQA (Yun et al., 2021)	✗	✓	✗	✓	✗	~5 s	Indoor/Outdoor
VideoEspresso (Han et al., 2025)	✗	✓	✗	✓	✗	N/A	Indoor/Outdoor
Ego4D (Grauman et al., 2022)	✗	✓	✓	✓	✗	~8 min	Indoor/Outdoor
EgoTracks (Tang et al., 2023)	✗	✗	✓	✓	✓	~6 min	Indoor/Outdoor
EPIC-KITCHENS (Damen et al., 2022)	✗	✗	✓	✓	✓	~7.5 min	Indoor
ObjChangeVR-Dataset (<i>ours</i>)	✓	✓	✓	✓	✓	~60/180 s	Indoor/Outdoor

Table 8: Comparison of the ObjChangeVR-Dataset with representative 3D and video-based benchmarks.

the MLLM. Our method incurs additional computation, mainly from the reasoning stage, since it requires processing multiple cross-view and cross-time frames. The total inference time remains under 10s.

A.7 Statistical Robustness of Evaluation Results

To assess whether the observed performance differences are robust to dataset size, we conduct a 1,000-sample bootstrap resampling analysis over all question-answer pairs. The 95% confidence intervals (CI95%) for EM@0.8 across all methods are shown in Table 12. The tight intervals indicate low variance and show that the performance differences (especially the gains of ObjChangeVR) are statistically robust.

A.8 Question-answer Pair Generation Prompt

To automatically generate disappearance reasoning question-answer pairs, we provide the MLLMs with a prompt that specifies the temporal relationship between two frames of the same VR scene.

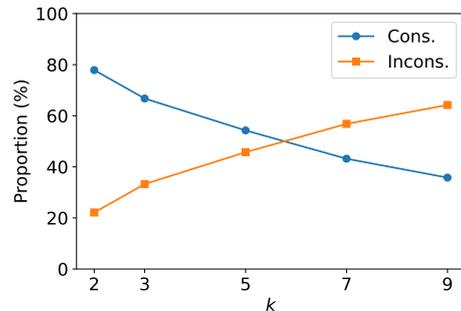


Figure 3: Proportion of questions (out of 5,000) with consistent and inconsistent intermediate answers across different k .

The first image represents the *previous* frame, while the second denotes the *current* frame. The model is instructed to generate ten question-answer pairs by comparing the two frames and reasoning about the presence or disappearance of objects over time.

The prompt enforces several key rules to ensure reasoning consistency. Questions must use environmental spatial anchors (e.g., “the chair near the window”). Each set of ten questions contains four

Method	Caption-CLIP	Image-CLIP	Viewpoint-Retrieval	CoT-SC	ObjChangeVR w/o TCV	ObjChangeVR
EM@0.70	0.5302	0.5934	0.6026	0.6958	0.6840	0.7588
EM@0.75	0.5302	0.5934	0.6026	0.6958	0.6840	0.7566
EM@0.80	0.5286	0.5924	0.6014	0.6950	0.6828	0.7540
EM@0.85	0.5214	0.5860	0.5926	0.6774	0.6656	0.7368
EM@0.90	0.5080	0.5678	0.5738	0.6626	0.6508	0.7182

Table 9: Sensitivity analysis of EM@ τ under different similarity thresholds.

Method	EM@0.8	Macro-F1	Weighted-F1
ObjChangeVR	0.8200	0.8070	0.8307

Table 10: Performance of ObjChangeVR on the object addition dataset.

about disappearing objects, three about objects that are present across both images, and three about objects that never appear in the scene. This balanced composition is designed to ensure sample diversity and prevent category bias during dataset generation. When fewer than four disappearing objects exist, the model is encouraged to rephrase similar queries with varied sentence structures to enhance linguistic diversity. The instruction provided to the model is shown in Table 13. An example of the output is shown in Table 14.

A.9 Case Study

To clearly illustrate the reasoning process of ObjChangeVR, we present two representative cases: one with consistent intermediate reasoning and another with inconsistent intermediate reasoning. The consistent intermediate reasoning case is shown in Table 15, while the inconsistent one is shown in Table 16. These examples provide an intuitive understanding of how ObjChangeVR aggregates multiple intermediate reasoning answers and obtains the final decision.

Method	Caption-CLIP	Image-CLIP	Viewpoint-Retrieval	CoT-SC	ObjChangeVR w/o TCV	ObjChangeVR
Retrieval (s)	0.035	0.261	0.003	0.004	0.004	0.003
Reasoning (s)	2.383	3.372	3.194	8.249	2.814	9.562
Total (s)	5.887	3.653	3.323	8.256	2.821	9.566

Table 11: Inference latency breakdown of different methods.

Method	CI95% (EM@0.80)
Caption-CLIP	[0.5140, 0.5424]
Image-CLIP	[0.5788, 0.6064]
Viewpoint-Retrieval	[0.5886, 0.6158]
CoT-SC	[0.6812, 0.7076]
ObjChangeVR w/o TCV	[0.6696, 0.6950]
ObjChangeVR	[0.7428, 0.7670]

Table 12: 95% confidence intervals (CI95%) of EM@0.80 obtained via 1,000-sample bootstrap resampling over all question-answer pairs.

[Instruction]

You are given two images of the same VR scene taken at different times.

The first image: previous screenshot.

The second image: current screenshot.

Your task:

Generate 10 simple question-answer pairs comparing the images, focusing on object disappearance or persistence.

Rules:

- Use spatial references (e.g., "the chair near the window"). The reference should be the environment such as the left and right walls, windows and arches, rather than specific objects.
- If an object was visible in the first image but is missing in the second image, answer: "It has disappeared."
- If an object was present in both, answer: "It has always been here."
- If an object was missing in the first image, answer: "It was never there."
- Each time, ask 4 disappearing objects, 3 always present objects, and 3 objects that have never been in the scene. (If there are fewer than four disappearing objects in the scene, you can repeatedly ask questions about the same disappearing object using different sentences.) But try as hard as possible to ask questions about different objects.
- Only questions about whether an object existed in the past are allowed. Only questions like "Did a specific object exist in a past scene?" are allowed. No other content can be asked. (When asking questions, never mention the two pictures and do not use words like "the differences or changes between the two pictures".) Use the past tense.
- When asking questions, the tone should seem as if you can only see the second picture (that is, the current picture), so when referring to relative positions, do not use objects that existed in the previous picture but not in the current one as references. When answering, you can see the information of the two pictures.
- Respect facts, not illusions.

Output Format:

1. <question> Answer: <answer>

2. <question> Answer: <answer>

Table 13: Instruction of question-answer pair generation.

[Input]



[Output]

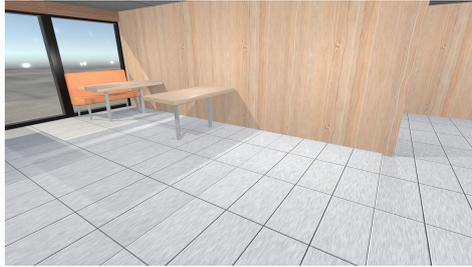
1. *Was there a round wire plant decoration near the left wall in the past?*
Answer: It has disappeared.
2. *Was there a small green cactus on the right side of the shelf in the past?*
Answer: It has disappeared.
3. *Was there a second cactus next to the first one on the right side of the shelf in the past?*
Answer: It has disappeared.
4. *Was there a thin white rectangular picture beside the deer painting near the right wall in the past?*
Answer: It has disappeared.
5. *Was the framed deer picture near the right wall here before?*
Answer: It has always been here.
6. *Was the small bird-shaped decoration on the right shelf here before?*
Answer: It has always been here.
7. *Were the books stacked on the lower right shelf here before?*
Answer: It has always been here.
8. *Was there a coat rack near the right wall in the past?*
Answer: It was never there.
9. *Was there a lamp near the left side of the scene in the past?*
Answer: It was never there.
10. *Was there a hanging decoration under the ceiling beams in the past?*
Answer: It was never there.

Table 14: An example of question-answer pair generation.

[Input]

<Question> *Were there some food on the tables near the right wall?*

<Current Frame>



<Retrieved Frames>



(a) Frame 41



(b) Frame 42



(c) Frame 43

[Output]

<Intermediate Reasoning Answer 1> In the retrieved picture, you can see food items on the tables near the right wall, specifically on the tables with orange chairs. In the current picture, those tables near the right wall are empty, with no food items present. So it has disappeared.

<Intermediate Reasoning Answer 2> In the retrieved picture, there is food on the tables near the right wall. In the current picture, those tables are empty with no food present. So it has disappeared.

<Intermediate Reasoning Answer 3> In the retrieved picture, there is food on the tables near the right wall. In the current picture, the tables near the right wall have no food on them. So it has disappeared.

<Final Answer> It has disappeared. Because all three sub-answers consistently indicate that there was food on the tables near the right wall in the retrieved images, but the current image shows those tables as empty. This consistency across different retrieved images suggests that the food was indeed present before and has since disappeared.

Table 15: A case of consistent intermediate reasoning.

[Input]

<Question> *Was there a drink dispenser on the left counter?*

<Current Frame>



<Retrieved Frames>



(a) Frame 64



(b) Frame 71



(c) Frame 147

[Output]

<Intermediate Reasoning Answer 1> In the retrieved picture, you can clearly see a drink dispenser labeled "Mr. Right" on the left counter. In the current picture, that drink dispenser is no longer there. So it has disappeared.

<Intermediate Reasoning Answer 2> In the retrieved picture, there is a drink dispenser on the left counter. In the current picture, that drink dispenser is no longer there. So it has disappeared.

<Intermediate Reasoning Answer 3> In the retrieved picture, there is no drink dispenser visible on the left counter. Similarly, in the current picture, there is still no drink dispenser visible on the left counter. So it was never there.

<Final Answer> It has disappeared. Because the drink dispenser labeled 'Mr. Right' is visible on the left counter in the first and second retrieved images but not in the third retrieved image or the current image. The inconsistency in the third sub-answer may be due to a different shooting time that failed to capture the dispenser. This suggests that the drink dispenser was indeed there before but has since disappeared.

Table 16: A case of inconsistent intermediate reasoning.