MirroS

S-Space: Exploring Spatial Workspace in Multimodal Models

MirroS Technical Report

Abstract

Multimodal models can describe spatial relations, yet the internal representations and mechanisms that support these abilities remain poorly understood. We investigate S-Space, a low-dimensional spatial workspace in which object-token representations encode continuous, viewer-centered coordinates along horizontal, vertical, and distance axes. Across multimodal models, this workspace supports linear readouts that generalize across datasets and prompting contexts, targeted interventions that causally alter spatial judgments with limited effects on unrelated attributes, and contextual steering and refinement of its coordinates. Beyond visible objects, S-Space also represents off-screen entities and abstract concepts within a shared spatial geometry. However, this shared geometry can entangle physical and abstract spatial meanings, coupling distinct reference frames and inducing errors in spatial reasoning. Even when spatial information is accurately represented, models often struggle to compute with it: explicitly transforming S-Space coordinates substantially improves perspective-taking over native answer generation. Together, these findings reveal both the capabilities and limitations of an internal spatial workspace, illustrating how mechanistic interpretability can guide the development of physical intelligence.

Date: September 7, 2026

Website: https://mirros.ai/blog/s-space

Code: https://github.com/mirros-lab/s-space

To understand space is to construct it within; to reason is to manipulate what we construct.

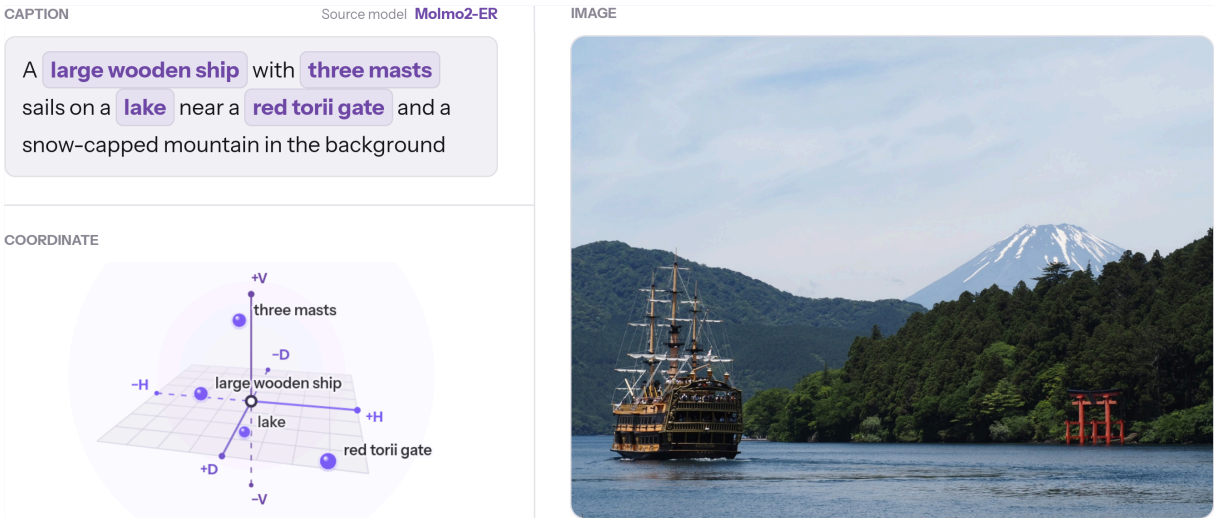

Figure 1 Exploring S-Space. Interactively select a case and an object phrase to see where it lies in S-Space. For more examples and interactive demos, visit our official website.

1 Introduction

At a glance, humans perceive a scene not merely as a collection of visible objects, but as a space organized around us. A cup is represented not only by what it is, but by where it lies relative to us—left or right, above or below, near or far. Underlying this seemingly effortless spatial perception, complementary neural systems help maintain a coherent spatial world: the *posterior parietal cortex* tracks where things lie relative to us, while the *hippocampal system* integrates changing views into a more stable map as we move through the environment [1]. What emerges is not necessarily a photorealistic replica of the world, but a structured internal geometry that cognition can work with: we can imagine the same room from another viewpoint, mentally move an object to a new location, and anticipate how the scene would change. **Spatial understanding, then, is more than perception: it is the construction of a manipulable workspace** in which objects are bound to locations and their relations can be maintained, transformed, and used for memory, reasoning, and action.

1.1 S-Space: Spatial Workspace

Multimodal AI systems have exhibited similar outward spatial behaviors. Given visual observations as images, vision-language models can identify objects, describe how they are arranged, answer spatial questions, and increasingly guide embodied agents through the physical world. Yet their observable answers and actions reveal little about the internal mechanisms supporting these abilities. A model might claim that one object is to the left of another by recognizing a local visual pattern or exploiting a language prior—or instead constructing a compact, coherent spatial representation of the world.

Our core team members in previous work [2] have caught *an early glimpse of such an internal spatial workspace*—what we refer to here as **S-Space**, illustrated in Figure 2. As a multimodal language model reasons through a two-dimensional visual grid world, its hidden activations implicitly track entity locations in the evolving world state, even when those coordinates are **never written out in the reasoning trace**. This result suggested a broader mechanistic picture: perception populates it with entity locations, ongoing computation maintains and updates them, and reasoning consults them as needed.

In this work, we dive deeper into how multimodal models represent space by studying **S-Space for three-dimensional real-world scenes**. We show that multimodal models perceive 3D spatial information from visual observations and write it into S-Space, making it available to downstream reasoning—even though, as we will see, models do not always use proficiently. Specifically, the *continuous* spatial coordinates of objects—and, strikingly, some abstract concepts—are encoded by linear components [3] of the models' activations at certain layers. This spatial subspace is persistently observed across a range of contexts, and can be manipulated to causally intervene in the model's spatial understanding. By measuring and manipulating these representations, we gain insights into how S-Space emerges from training (§4), how it is accessed during inference (§5), and how its contents evolve as test-time computation scales (§6).

1.2 Takeaways

1. **Multimodal models may already carry an internal map of the spatial world.**
   S-Space encodes continuous 3D object locations (§3.1), persists across prompts and contexts (§3.3), and can be causally manipulated to change the model's spatial judgments (§3.2).
2. **S-Space extends beyond what the camera sees.**
   The observer (§3.4)—and even abstract concepts expressed through language—can occupy meaningful positions in the same spatial geometry, revealing a shared but sometimes entangled workspace for vision and language (§4.1).
3. **Models often know where things are before they know how to reason about them.**
   Chain-of-thought shows signs of internally transforming S-Space (§6.1), but the manipulation remains imperfect; explicit computation over the very same representation can substantially outperform the model's native reasoning (§5.1).
4. **S-Space is an evolving workspace, not a static snapshot.**
   Multimodal evidence and supervision can continually reshape and refine S-Space (§4.2, §6.2), pointing toward physical intelligence whose internal world representations are continually shaped by richer interaction with the world (§7).

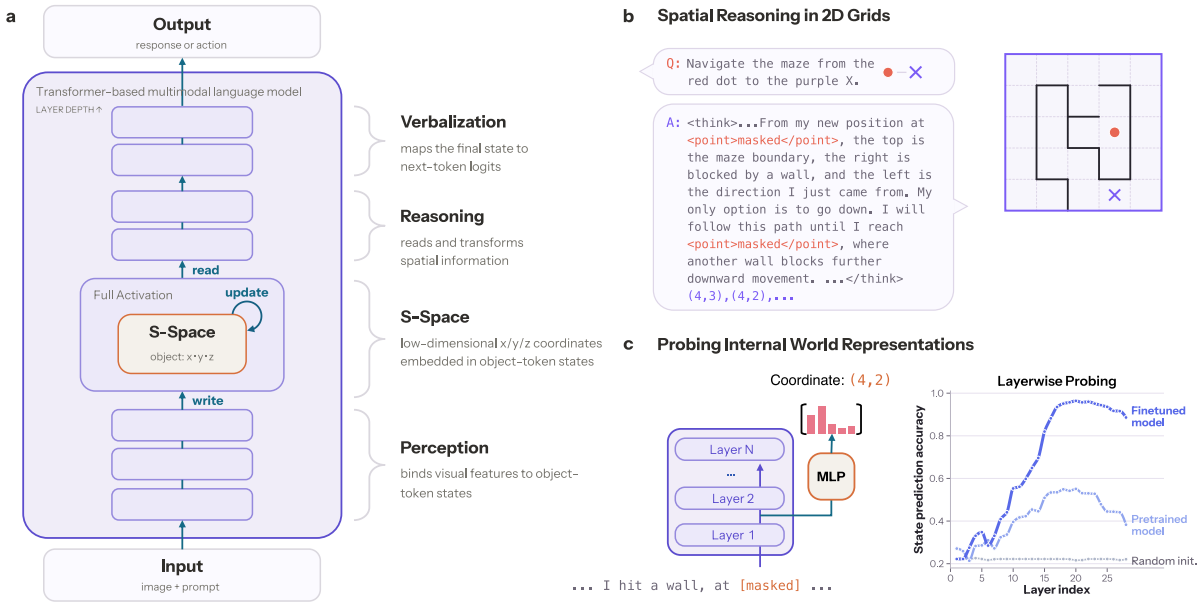

Figure 2 **The concept of S-Space. a,** S-Space is an internal spatial workspace in multimodal language models. Perception writes spatial information into S-Space, internal computation transforms and updates this information, and later layers read it to produce a response. **b-c,** Our previous work [2] reveals early evidence for S-Space in 2D grid worlds, showing that hidden activations track evolving 2D coordinates even when they are never spoken out in the reasoning trace.

2 Method

A transformer-based vision-language model processes an image and its accompanying text as a sequence of tokens. For each token, the model maintains a representation vector, forming a shared workspace that is repeatedly read from and written to across layers and successive token positions. As computation proceeds, these representations accumulate visual, linguistic, and relational information. At the final layer, the representation at the last position is mapped by the language-model head to logits over the vocabulary, from which the next token is predicted.

We search for S-Space in the model's intermediate layers, as early layers remain relatively close to the input representations, and later layers become increasingly aligned with next-token prediction and the production of a verbal response.

As illustrated in Figure 3, we hypothesize that an object's spatial coordinates in a viewer-centered, or egocentric, reference frame[1] are linearly composed into the representation at the last token t of its name [3]:

$$x = v_x^\top h_{\ell,t}, \qquad y = v_y^\top h_{\ell,t}, \qquad z = v_z^\top h_{\ell,t},$$

where $h_{\ell,t} \in \mathbb{R}^{d_{\text{model}}}$ is the representation at layer ℓ, and $v_x, v_y, v_z \in \mathbb{R}^{d_{\text{model}}}$ are readout directions for the horizontal, vertical, and distance axes, respectively.[2]

Concretely, to identify this three-dimensional subspace, we collect a large image corpus and construct prompts p that compare a query object q with a reference object r [4], for example, "`<image> Is the [object_qry] to the left or right of the [object_ref]?`". For each spatial axis g, we define a final-answer logit difference s_p—for example, $s_p = \text{logit}(\text{right}) - \text{logit}(\text{left})$ for the horizontal axis—and backpropagate it to the two object-token states at the source layer:

[1] As we will see later, this reference frame is not fixed: it can be steered by context and progressively reoriented over the course of chain-of-thought reasoning.

[2] For the horizontal and vertical axes, we use image-plane coordinates rather than world coordinates, without explicitly disentangling depth-dependent perspective effects. These projected coordinates suffice for most pairwise spatial questions considered here.

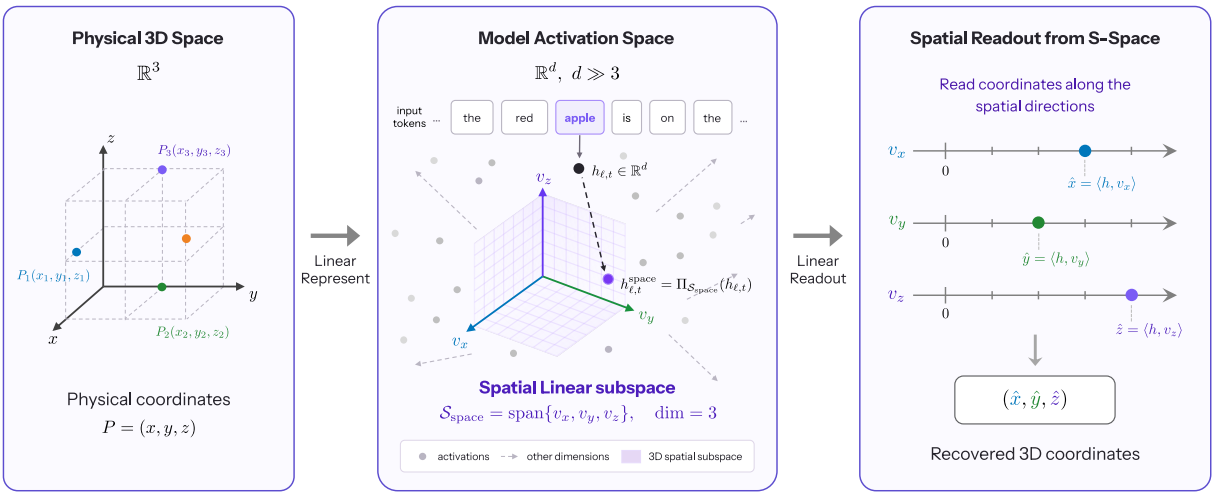

Figure 3 Linear spatial representation in S-Space. An object's spatial coordinates are linearly embedded in its high-dimensional token representation, where the horizontal, vertical, and distance directions form a three-dimensional S-Space. Reading along these directions recovers the object's viewer-centered coordinates.

$$v_g^{\text{logit}} = \text{unit}\left(\mathbb{E}_p\left[\frac{\nabla_{h_{\ell,q}} s_p - \nabla_{h_{\ell,r}} s_p}{2}\right]\right).$$

The role-antisymmetric difference suppresses components shared by the two objects, while averaging over prompt templates and original/swapped object orders reduces prompt-specific variation.

3 The Spatial World in S-Space

S-Space is a linear representation subspace that encodes the spatial layout of a scene along horizontal, vertical, and distance axes. In this section, we demonstrate three of its core properties: the accuracy with which it represents object locations, the extent to which it can be causally manipulated, and the persistence of its spatial structure across different contexts.

Datasets & Benchmarks. We identify S-Space using COCO images and object annotations [5], and evaluate it on three spatial question-answering benchmarks: CV-Bench [6], EmbSpatial-Bench [4, 7], and SpatialTunnel [4]. CV-Bench provides a partially in-distribution evaluation through its COCO-derived examples, while the latter two test out-of-distribution generalization. SpatialTunnel is a fully synthetic benchmark designed to eliminate the vertical-distance entanglement common in natural images.

Models. We analyze S-Space in Molmo2-ER, its downstream vision-language-action model MolmoAct2 [8], and general-purpose models from the Qwen family [9]. Molmo2-ER is particularly well suited to our analysis: it is fully open, specialized for spatial and embodied reasoning, and offers unusual transparency into its training data and recipe.

3.1 Accuracy

S-Space encodes continuous object coordinates. We project each object-token representation onto the spatial directions of S-Space and compare the resulting coordinates with ground-truth object positions. Across models and datasets, both individual object coordinates and pairwise coordinate differences strongly correlate with their ground-truth counterparts along the horizontal, vertical, and distance axes, as shown in Figure 4. S-Space therefore captures more than categorical relations such as left or right: it preserves a continuous geometry of object locations.

S-Space directly supports spatial QA. We next answer spatial-relation questions using only the coordinates read from S-Space: the sign of the coordinate difference determines whether one object is left or right, above or below, and nearer or farther than another. This simple coordinate-based readout achieves high accuracy across CV-Bench, EmbSpatial-Bench, and SpatialTunnel, including benchmarks

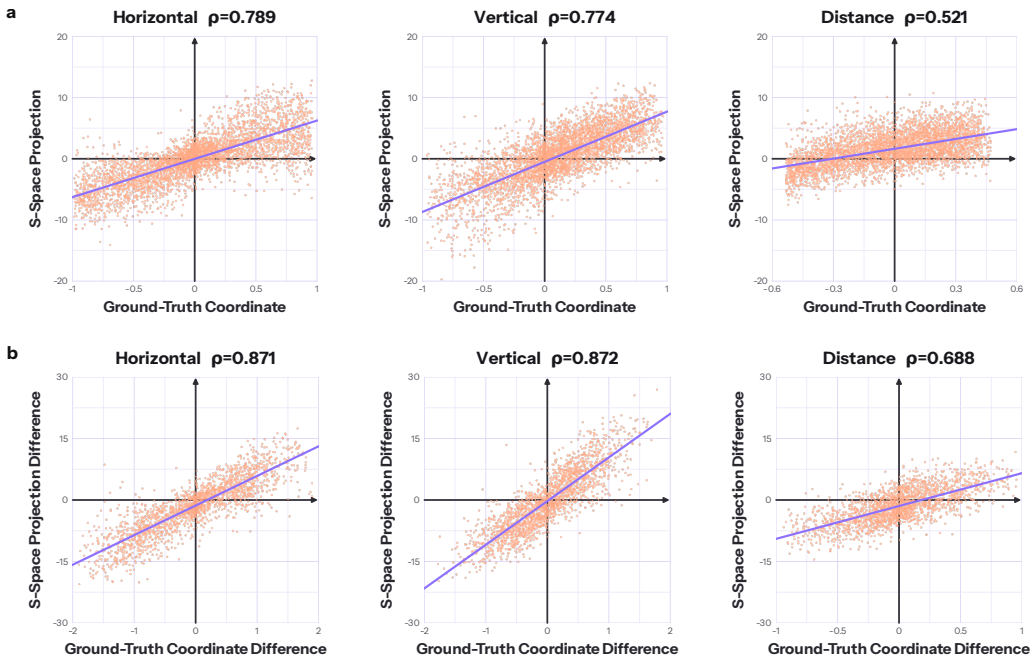

Figure 4 Coordinate correlation. Results for Molmo2-ER at layer 21 on a subset of COCO. **a,** Correlation between ground-truth object coordinates and their S-Space projections. **b,** Correlation between ground-truth pairwise coordinate differences and the corresponding differences in S-Space. ρ denotes the correlation coefficient; blue lines indicate linear fits.

outside the distribution used to identify S-Space. The result in Figure 5b shows that the geometry encoded in S-Space is sufficiently accurate to support spatial judgments directly.

3.2 Causality

Accurate readout does not necessarily mean that the model itself uses S-Space. To test its causal role, we intervene on object-token activations at the layers where S-Space is most prominent. We selectively modify an object's coordinate along the queried axis—for example, reversing the horizontal relation between two objects—while preserving the remaining components of their representations. We then continue the forward pass without changing the image or prompt.

Manipulating S-Space predictably changes spatial judgments. We intervene by shifting an object-token activation along the relevant S-Space axis to a counterfactual coordinate while leaving the remaining components unchanged. Manipulating S-Space predictably shifts the model toward the corresponding counterfactual answer, as illustrated in Figure 6: *left* becomes *right*, *above* becomes *below*, and *closer* becomes *farther*. We measure both the change in answer-logit margin and the fraction of successfully flipped predictions. These interventions show that S-Space is not merely readable, but causally involved in the model's spatial judgments.

Non-spatial judgments remain largely intact under S-Space interventions. The same interventions that strongly alter spatial answers have little effect on unrelated judgments, such as object color, as shown in Figure 7b. This selective effect suggests that S-Space interventions modify spatial information without broadly disrupting the object representation, strengthening the evidence that S-Space plays a specific causal role in spatial reasoning.

3.3 Persistence

S-Space is not imposed by our probing procedure, nor is it specific to the data or prompts used to extract it. The Accuracy experiments already show that coordinates extracted from one dataset generalize to spatial questions in others. We further replace the relative-comparison prompts used during extraction with independent single-object queries such as "`Where is the [object]?`" Each object is processed in

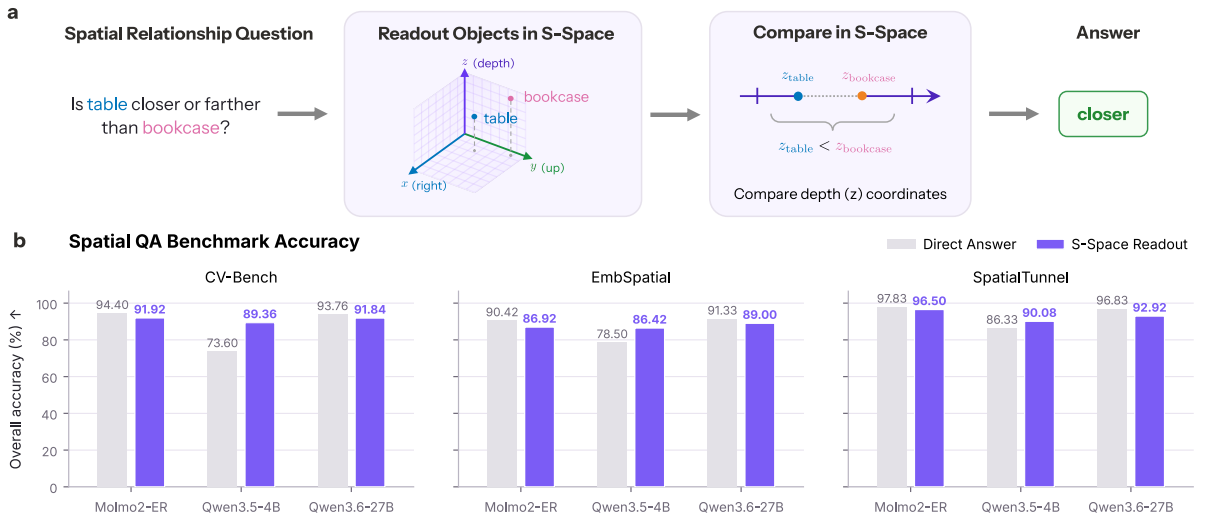

Figure 5 **Spatial QA directly from S-Space. a,** A spatial question can be answered by reading the queried objects' coordinates from S-Space and directly comparing them along the relevant axis. **b,** This simple coordinate-based readout achieves high accuracy across CV-Bench, EmbSpatial-Bench, and SpatialTunnel, comparable with the model's direct answers.

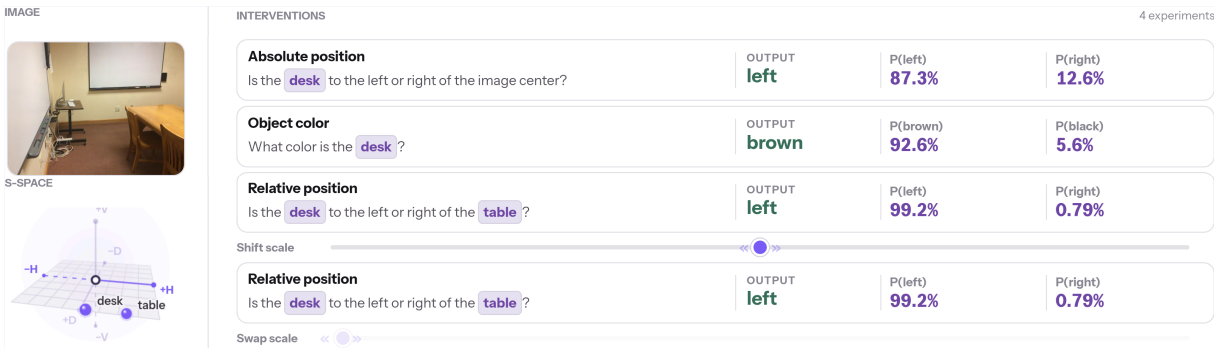

Figure 6 **Editing S-Space causally changes the model's output.** Interactively modify object coordinates in S-Space and observe how the model's answer probabilities respond across spatial and non-spatial questions. Each $P(\text{token})$ denotes the softmax probability assigned to the corresponding candidate answer token. For more examples and interactive demos, visit our official website.

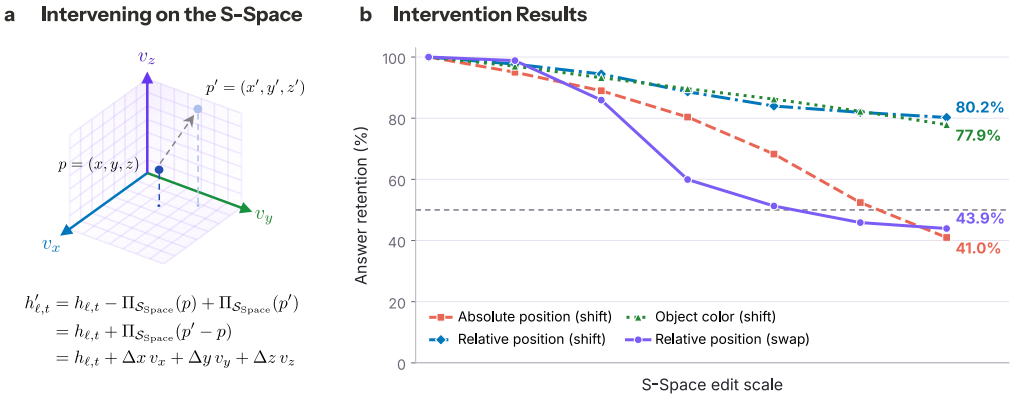

Figure 7 **Quantifying the specificity of S-Space interventions. a,** An object-token representation is edited by moving its encoded position from p to p' within S-Space while preserving components outside the spatial subspace. **b,** As the edit scale increases, judgments targeted by the intervention degrade substantially, whereas untargeted spatial judgments and object-color predictions remain largely preserved, demonstrating the specificity of the causal effect. The dashed line marks 50% answer retention.

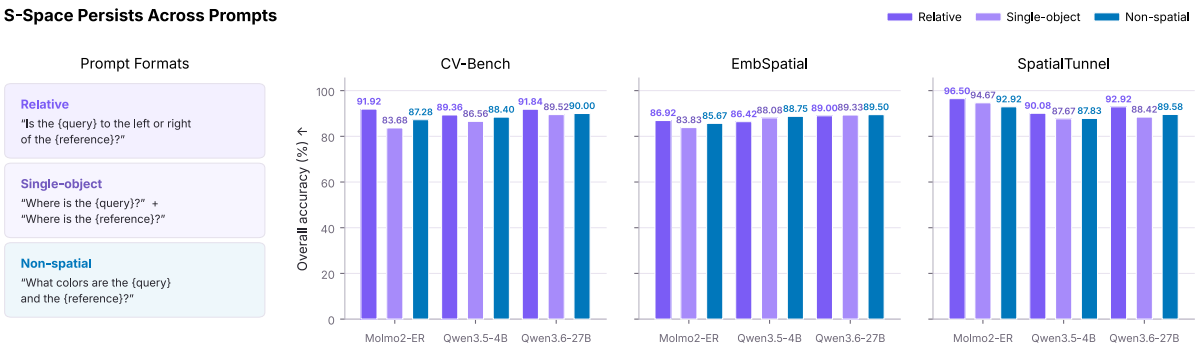

Figure 8 **S-Space persists across prompts.** The same fixed, model-specific S-Space readout continues to support accurate spatial judgments across pairwise spatial, single-object, and non-spatial prompts, suggesting that the representation is not induced by a particular prompting format.

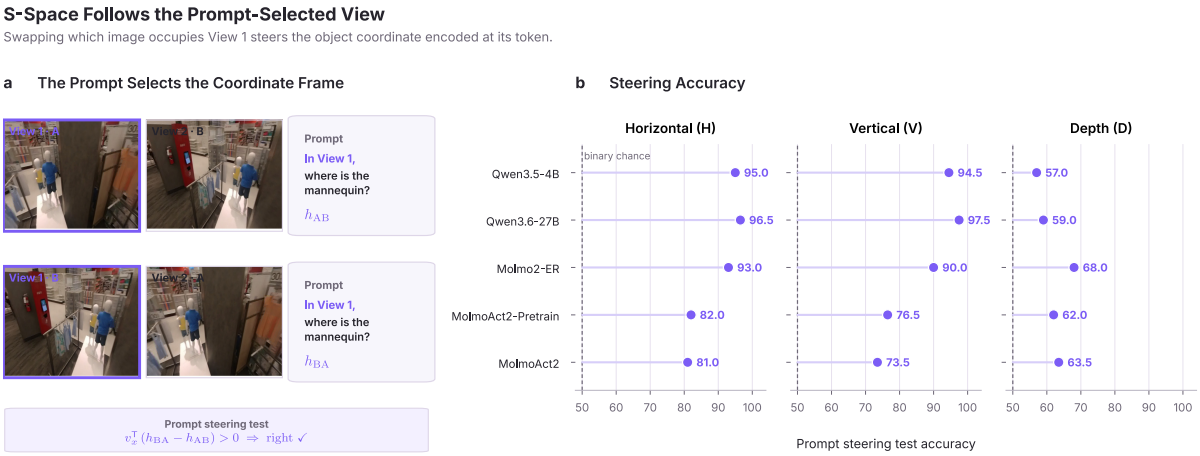

Figure 9 **S-Space follows the prompt-selected view.** Each example is evaluated twice with the same View-1 query while the two input images are swapped. We then compare the resulting change in the object-token coordinates with the object's ground-truth displacement between the two views.

a separate forward pass, and their extracted coordinates are compared only afterward. Finally, we test non-spatial questions that never ask about location, such as "`What colors are the [query] and the [reference]?`"

S-Space is intrinsic rather than induced by the extraction procedure. Despite this substantial change in prompt structure and computational context, the resulting coordinates continue to support accurate spatial QA, as shown in Figure 8. This robustness suggests that S-Space is an intrinsic, persistent organization of the model's representations rather than an artifact induced by a particular dataset, prompt template, or pairwise-query format.

S-Space generalizes and is steerable in multi-view contexts. Although S-Space is identified entirely from single-view examples, it still provides meaningful spatial coordinates when multiple views of the same scene are presented together. We further ask whether the model can use the prompt to select which view defines the observer-centered frame. As illustrated in the prompt-steering test in Figure 9, we keep the prompt fixed on "`In View 1`" while swapping which image is designated as View 1, and test whether the resulting change in an object's S-Space coordinates follows the corresponding change in viewpoint. Across models, steering accuracy is consistently well above random choice along the horizontal, vertical, and distance axes. These results suggest that S-Space is not tied to a single visual input, but can be dynamically anchored to the viewpoint selected through language.

3.4 Beyond the Visible

So far, we have described S-Space primarily as a representation of visible objects. But is its geometry limited to what the model can actually see? Intriguingly, we find that its geometry extends beyond objects directly observed in the image, as shown in Figure 10.

S-Space localizes off-screen entities. Even when an entity itself is not visible, its position can be inferred from spatial cues in the scene. A goalkeeper diving toward an unseen ball places the *ball* in the direction of the dive; a taut leash places the unseen *owner* beyond the dog; and railway tracks leading into a tunnel place an unseen *train* farther along the depth axis. These cases suggest that S-Space does not merely register detected objects, but can complete the surrounding spatial world from actions, physical relations, and scene geometry.

S-Space places the invisible observer. In prompts such as "`Compared with {ref}, is {subj} farther from or closer to you?`", the token corresponding to *you* projects particularly strongly onto the distance axis, precisely toward its *close* end. This arrangement is naturally consistent with an egocentric reference frame: *you* denotes the observer relative to whom near and far are defined, effectively anchoring the near end of the spatial axis despite never appearing in the image.

S-Space reflects relations implied by language. Holding the image fixed while changing only a spatial relation in the prompt can move the corresponding coordinates in S-Space. For example, describing a delivery robot as moving *past you* versus moving *toward you* reverses the inferred relative depth of the invisible *you*, while leaving the robot's position largely unchanged. S-Space therefore incorporates not only visual evidence, but also spatial relations implied purely through language.

Together, these observations suggest that S-Space is not merely a visual map of detected objects, but a shared geometry in which visually grounded entities and linguistically specified concepts can occupy meaningful spatial positions. This, in turn, raises a deeper question: how does S-Space emerge, bringing visual objects and linguistic concepts into a common spatial workspace?

4 How Does S-Space Emerge?

Multimodal AI has not directly encountered the physical world, but learns from partial projections of the world, such as images, language, video, and action trajectories. The Platonic Representation Hypothesis [10] suggests that, by learning to predict these observations, models may recover shared structure in the underlying world—S-Space can be viewed as one such geometry emerging through training. Yet what S-Space captures is also bounded by the projections available, motivating us to examine how different training modalities give rise to—and constrain—this spatial workspace.

4.1 Language Supervision Blends Physical and Abstract Concepts

We hypothesize that S-Space emerges partly from the objective of predicting spatial language during pretraining. To predict tokens such as *left* and *right*, a VLM must bind visible entities to their relative positions and make this information accessible to language generation. This pre-training pressure may organize internal activations into stable spatial axes without explicit coordinate supervision. This same learning process may carry the broader structure of language, including metaphorical associations and competing reference frames, into the model's spatial reasoning.

Political "left" maps onto physical left in S-Space. Language is a highly abstract modality for representing the world. It not only describes physical geometry but also repurposes spatial terms to express abstract meanings. Models that inherit this semantic ambiguity struggle to cleanly disentangle physical locations from abstract associations. Strikingly, *socialist*—a concept associated with the political left—also projects toward the left end of the horizontal axis, as shown in Figure 11. This does not necessarily mean that the model literally places an ideology in physical space; rather, it suggests that the physical and metaphorical meanings of *left* share internal representational structure.

Cardinal and image-relative directions are coupled in S-Space. Such coupling is not always benign: it can directly induce reasoning failures. In cardinal-direction spatial QA, we repeatedly observe the model inferring cardinal directions from image-relative relations without first resolving the reference frame—for example, naively mapping right to east. In one case, although the question establishes a north

IMAGE

S-Space places the unseen ball in the goalkeeper's diving direction.

PROMPT

The **goalkeeper** dives toward the **ball**.

S-SPACE READOUT — H-axis

ball	-0.94 [-1.17, -0.69]
goalkeeper	+1.42 [+1.33, +1.50]

left -3.00 — 0 — +3.00 right

IMAGE

S-Space places the unseen owner along the direction of the dog's taut leash.

PROMPT

The **dog** pulls toward its **owner**.

S-SPACE READOUT — H-axis

owner	+1.46 [+1.11, +1.80]
dog	+0.29 [-0.25, +0.78]

left -3.00 — 0 — +3.00 right

IMAGE

S-Space places the unseen train farther down the tracks than the worker.

PROMPT

The **worker** waits for the **train** beside the tunnel.

S-SPACE READOUT — D-axis

train	+1.63 [+1.24, +2.04]
worker	+3.49 [+3.06, +4.01]

far -3.00 — 0 — +5.00 close

IMAGE

Even without visual grounding, S-Space places "you" at the observer end of the depth axis.

PROMPT

Compared with **curtains**, is the **desk** farther from or closer to **you**?

S-SPACE READOUT — D-axis

you	+4.20 [+3.00, +5.32]
desk	+2.15 [+2.02, +2.29]
curtains	-1.52 [-1.62, -1.42]

far -3.00 — 0 — +8.00 close

IMAGE

S-Space reads "past" as placing "you" farther from the robot; other words shift the placement.

PROMPT

Moving past **you**, the **delivery robot** continues along its route.

S-SPACE READOUT — D-axis

you	+2.60 [+2.34, +2.86]
delivery robot	+2.97 [+2.92, +3.02]

far -1.00 — 0 — +5.00 close

IMAGE

Changing "past" to "toward" instead places "you" closer to the robot.

PROMPT

Moving toward **you**, the **delivery robot** continues along its route.

S-SPACE READOUT — D-axis

you	+4.59 [+4.39, +4.79]
delivery robot	+2.98 [+2.91, +3.04]

far -1.00 — 0 — +5.00 close

Figure 10 Beyond visible objects. Scenes and representative prompts used to probe off-screen entities, observer, and language-implied relation. Readouts show the mean coordinate and 95% prompt-template bootstrap confidence interval.

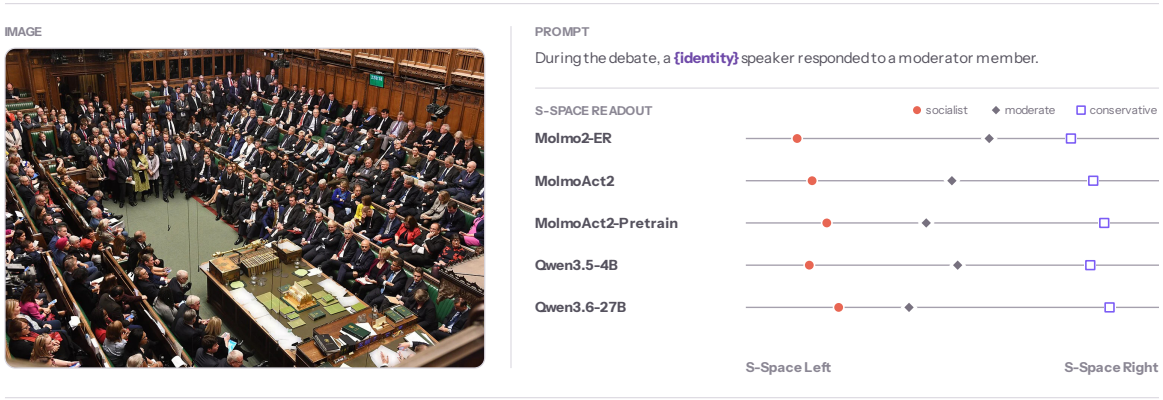

Figure 11 Political semantics align with the horizontal axis of S-Space. Across models, *socialist* projects toward S-Space Left, *conservative* toward S-Space Right, and *moderate* lies in between. Horizontal-axis readouts for the three identity tokens are z-scored independently within each model; the image and prompt are held fixed.

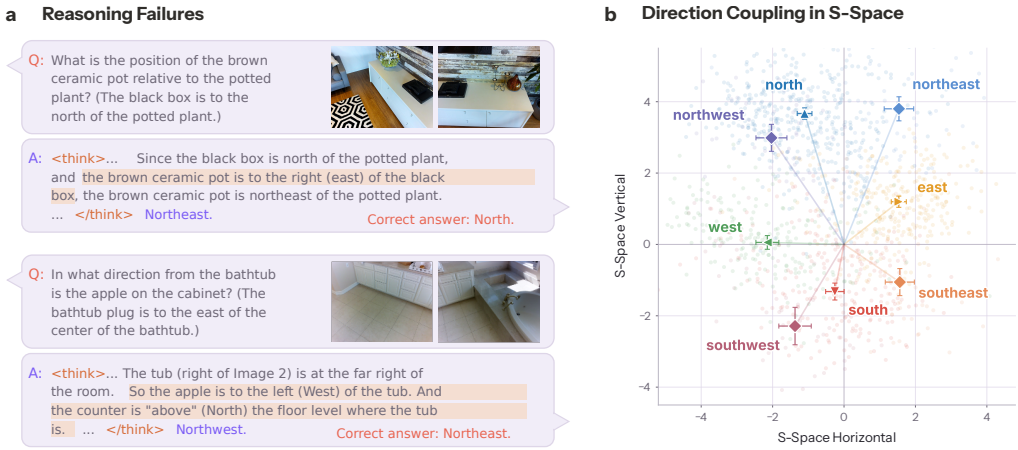

Figure 12 Direction coupling in S-Space. a, Representative reasoning failures where image-relative relations are incorrectly reused as cardinal directions. **b,** S-Space coordinates of cardinal-direction tokens from Qwen3.6-27B chain-of-thought on MMSI-Bench. Cardinal directions align with the corresponding image-relative axes in a map-like convention: north/up, south/down, west/left, and east/right. This reveals an internal coupling that can induce the failures in a.

relation, the model additionally interprets a rightward relation as east and consequently answers northeast instead of north. We suspect that this coupling may partly reflect biases in web-scale pretraining data, where cardinal directions frequently appear in map-like reference frames aligned with horizontal and vertical axes.

S-Space allows us to inspect the representational mechanism underlying this failure. Using Qwen3.6-27B on the object/region–object/region relationship tasks of MMSI-Bench [11], we collect cardinal-direction tokens (north, south, etc.) from the model's chain-of-thought and read out their coordinates in S-Space. The resulting geometry makes the coupling explicit, as illustrated in Figure 12b: east projects toward S-Space Right, north toward S-Space Up, while northeast combines both components. This suggests that the model represents cardinal and image-relative directions through partially shared spatial features rather than fully independent concepts. During cardinal-direction reasoning, this pre-existing alignment can therefore cause an image-relative cue to be reused as a cardinal one, producing the observed failures.

4.2 Action Supervision Sharpens Interaction Geometry

Language translates space into words. Action, by contrast, translates space into movement, providing a more precise interface with the physical world. To reach, grasp, or manipulate an object, a model must

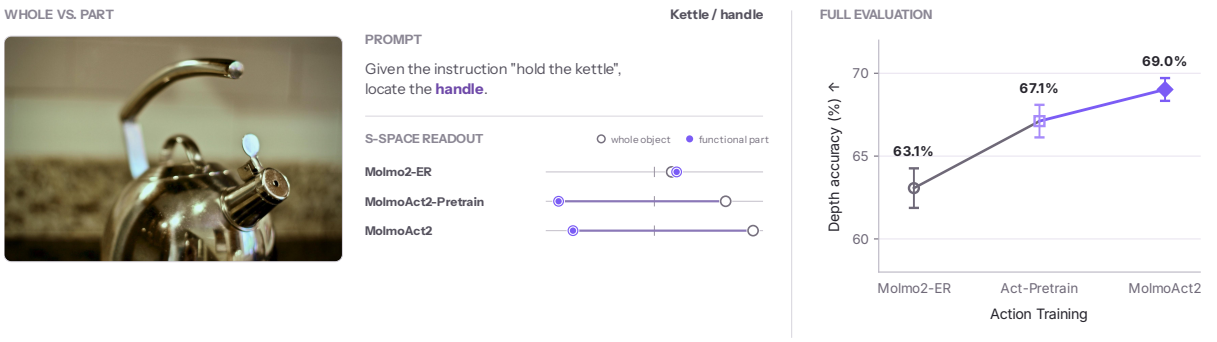

Figure 13 **Action supervision sharpens part–whole depth geometry in S-Space. Left**: Examples showing S-Space depth readouts for a whole object and its functional part. **Right**: Evaluation reveals that the action-supervised model makes more accurate part–whole depth judgments than the pure VLM. Error bars indicate 95% CIs across prompts.

resolve both the geometry of the scene and the functional parts through which interaction occurs. We therefore ask whether learning to act sharpens the geometry in S-Space by comparing a VLM with its fine-tuned VLA models.

Learning to act sharpens the part–whole geometry of S-Space. We examine images from the InstructPart dataset [12], in which a manipulable object contains a characteristic functional part with estimated depth of the part and the whole object. We construct prompts such as "`... Locate the [object]`" and assess the S-Space readout on the depth axis by distance comparison between the part and the whole, as in §3.1. We find an orderly progression along the model family: from the VLM through the action-model pretrain to the VLA, this local geometry becomes increasingly faithful to the physical relation between them. Learning to act appears to do more than shape a model's behavior; it also brings the spatial workspace underlying that behavior into finer focus.

5 Is S-Space Proficiently Used By Internal Reasoning?

In §3, we have established that multimodal models write spatial information into S-Space. A workspace is useful only insofar as its contents are accessible and used in downstream computation. Humans can mentally rotate and manipulate their internal spatial representations. We therefore ask whether models can proficiently use the spatial information encoded in S-Space to solve spatial reasoning tasks.

5.1 Failure Analysis in Spatial Reasoning

We study perspective-taking tasks from SpinBench [13], beyond simple spatial relation tasks studied in §3.1. Given a scene viewed from the front, the model must imagine and determine the spatial relationship of two objects when observing them from the left, right, or back. Yet frontier VLMs perform surprisingly poorly on these seemingly simple transformations. This raises a natural question: does the failure arise from spatial perception, or from the computation performed over the perceived spatial representation? To disentangle the two, we construct a decomposed pipeline, as shown in Figure 14b, that reads the objects' relative coordinates from S-Space, applies a fixed rotation corresponding to the target viewpoint, and derives the answer with a deterministic spatial rule. We compare this method with the model's native answer generation, both with and without chain-of-thought reasoning.

Explicit computation over S-Space substantially improves perspective-taking. Molmo2-ER achieves 60.96% accuracy through direct generation, compared with 94.52% using the prespecified S-Space readout and external rotation, as shown in Figure 14c. This gap indicates that spatial information encoded within the model can support substantially better performance when the required coordinate transformation is performed proficiently.

Chain-of-thought does not fully recover the missing spatial computation. With chain-of-

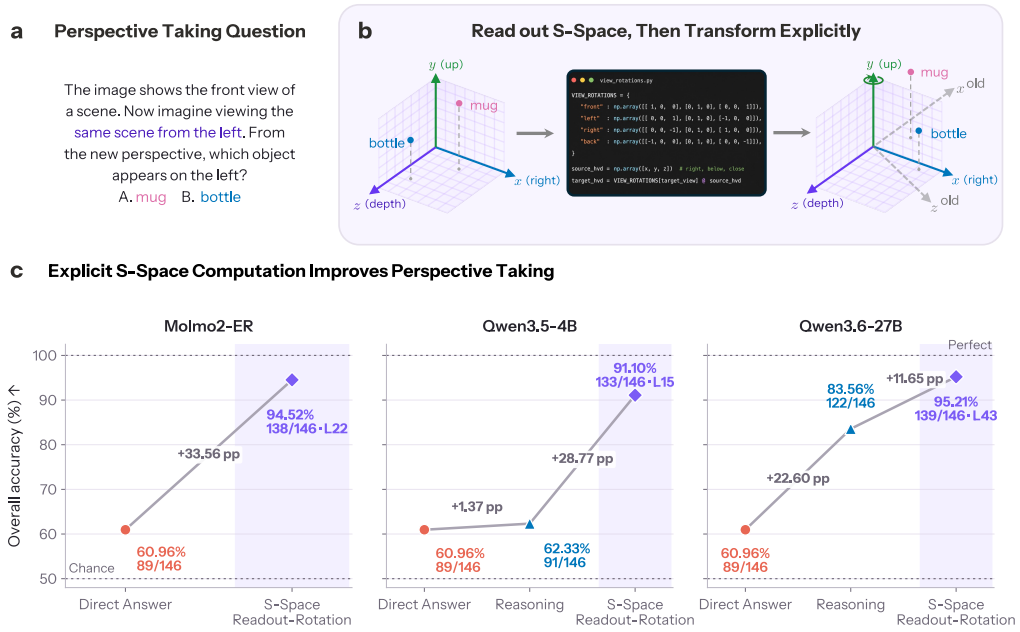

Figure 14 Decomposing perspective taking with S-Space. a-b, We separate spatial perception from spatial computation by reading object coordinates from S-Space, applying the viewpoint transformation explicitly, and deriving the answer with a deterministic spatial rule. **c,** Across perspective-taking tasks from SpinBench, this decomposed S-Space computation substantially outperforms direct generation and, where applicable, chain-of-thought reasoning, indicating that the models' internal spatial representations are more capable than their native spatial computation.

thought reasoning, Qwen3.6-27B reaches 83.56%, yet errors persist despite substantially increased test-time computation. Language-based reasoning can therefore improve performance, but does not guarantee reliable operations over spatial representations (see also §6.1).

Overall, these results suggest a perception–computation decomposition of spatial reasoning: VLMs can form capable spatial representations, yet fall short in the spatial computation required to answer reasoning questions correctly.

6 Can S-Space Evolve at Test Time?

So far, we have examined what S-Space represents and how well models use it. We next turn to another important question: is S-Space itself dynamic at test time? Modern agents commonly scale at test time along two complementary dimensions. **Computation scaling** allows the model to reason longer through chain-of-thought, while **context scaling** enriches its inputs with additional observations, retrieved knowledge, or dynamically loaded skills [14]. Our multi-view experiments in §3.3 already provide an early indication of this flexibility: S-Space can be steered toward the view specified by the prompt. We now ask how far this dynamics extends—can reasoning transform S-Space, and can new context refine it?

6.1 Computation Scaling: Transforming S-Space through Reasoning

As shown in Figure 2, our previous work [2] showed that multimodal models can track evolving two-dimensional coordinates within an internal workspace as the chain-of-thought unfolds. Here, we ask whether the three-dimensional representations in S-Space can likewise be updated and manipulated. We trace the S-Space coordinates associated with relevant object mentions throughout Qwen's reasoning on perspective-taking questions.

Specifically, as shown in Figure 15a, we construct a formulated chain-of-thought that explicitly progresses from the original spatial relation to the target viewpoint and then to the transformed relation. By probing repeated object mentions across these stages, we test how their S-Space coordinates update accordingly.

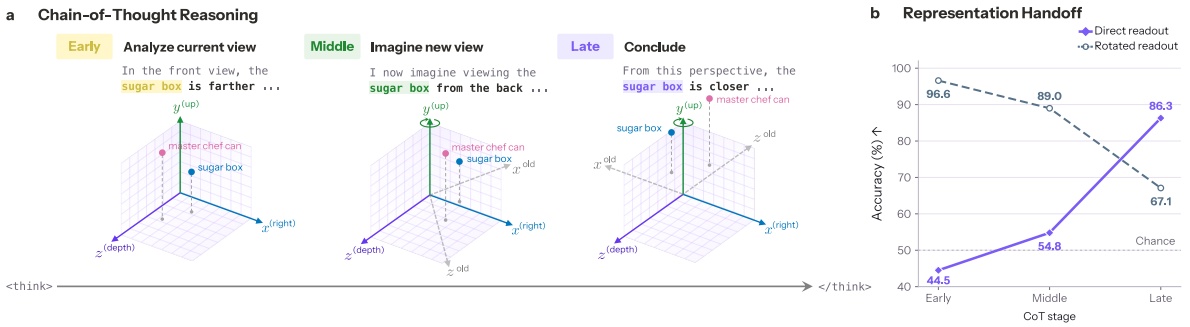

Figure 15 **Chain-of-thought progressively transforms S-Space. a,** S-Space coordinates of repeated object mentions gradually rotate from the original viewpoint toward the imagined target viewpoint over the course of reasoning. **b,** the corresponding readout advantage shifts from the rotated readout at early stages to the direct readout at late stages, suggesting a partial internalization of the perspective transformation.

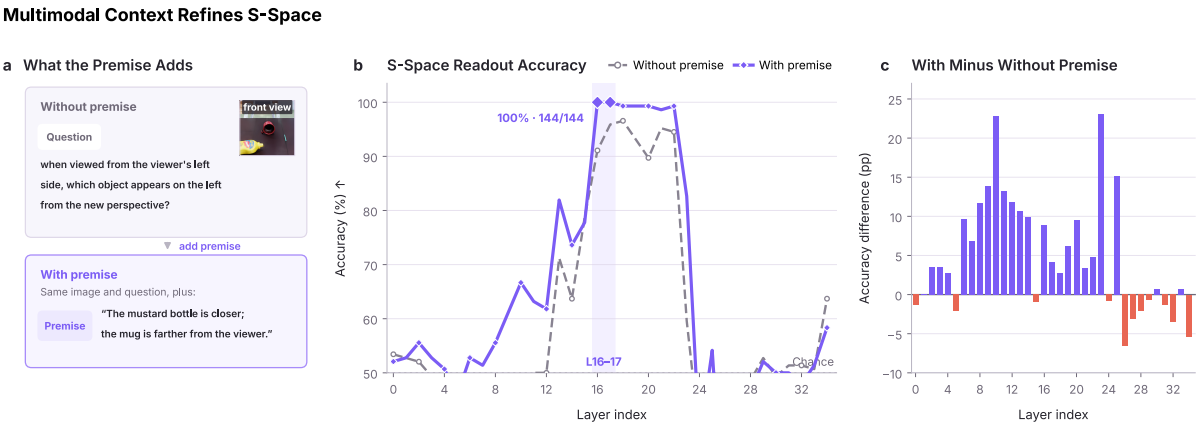

Figure 16 **Multimodal context refines S-Space. a,** The with-premise setting augments the visual observation with a textual description of the objects' spatial relation in the original view. **b-c,** Adding this premise improves S-Space readout accuracy across intermediate layers, reaching near-perfect accuracy at the strongest layers.

At each stage, we read out the S-Space coordinates and use them to answer the corresponding SpinBench perspective-taking question, either directly or after applying the optimal viewpoint rotation, as in §5.1.

Reasoning progressively transforms S-Space. As the chain-of-thought unfolds, the objects' relative coordinates become increasingly aligned with the viewpoint requested by the question. This transformation is reflected in a progressive shift from an advantage for the rotated readout at earlier stages to an advantage for the direct readout at later stages, as shown in Figure 15b. This suggests that the model internally simulates part of the required perspective transformation. However, this transformation remains imperfect and does not always produce the correct answer, consistent with the limitations identified in the preceding failure analysis (§5.1).

6.2 Context Scaling: Refining S-Space with Multimodal Evidence

We next simulate context scaling using the *with-premise* setting of SpinBench, where the image is supplemented with a textual description (the *premise*) of the objects' spatial relation from the original viewpoint. This provides additional spatial evidence in a modality different from visual observation.

Extended context refines S-Space. With the premise included, the coordinates read from S-Space become more accurate, allowing the readout-and-rotation pipeline to answer perspective-taking questions more reliably, as shown in Figure 16b–c. This suggests that S-Space is not fixed after visual perception: new contextual evidence can be incorporated into the workspace and refine its spatial representation.

7 What's Next

Physical Intelligence Development Guided by Mechanistic Interpretability

Spatial understanding and reasoning are fundamental to human cognition and to agents operating in the physical world. Yet scaling data and model size alone has proven insufficient: even frontier multimodal models consistently fail on basic visual tasks that humans, including three-year-old children, can solve effortlessly [15].

S-Space helps us understand this gap more deeply, allowing us to distinguish failures of perception from failures of computation. This illustrates how mechanistic interpretability methods [16] can serve as scientific instruments for understanding, monitoring, and guiding the development of frontier intelligence. Rather than evaluating models only through their outputs, mechanistic approaches break models into components whose functions are easier to interpret and characterize how those components interact to produce the whole model's behavior.

Our team has a longstanding interest in analyzing the internal representations and mechanisms of multimodal models. Our goal is to **uncover which key capabilities emerge through scaling — and which do not** — in multimodal intelligence, and, for those that do not, determine their requirements for new training signals, computational mechanisms, or experience with the physical world.

The insights revealed by S-Space point toward our next steps for building general multimodal intelligence in the physical world:

- **From spatial perception to agentic physical reasoning**. VLMs can form capable spatial representations, yet fall short in the spatial computation required to answer reasoning questions correctly. Future systems need not rely on a monolithic multimodal model to perform every spatial operation and physical reasoning internally. Agentic spatial reasoning [17] can connect learned spatial representations with dedicated mechanisms for coordinate transformation, simulation, memory, planning, and tool use—turning capable perception into reliable reasoning and action.
- **Intelligence evolving in the physical world**. Modalities through which a model learns are its windows onto the world, shaping both the structure and the limits of what it can represent. Physical intelligence should not be learned from static images and language alone, but from multimodal experiences generated through interaction with the world—including video, action, proprioception, touch, and environmental feedback. Omni-models that incorporate action [18] represent one instance of this broader direction, not its endpoint.

Ultimately, building physical intelligence requires more than scaling what models see, say, and do. It also requires understanding—and deliberately shaping—the internal representations and mechanisms through which they perceive, reason, and act.

References

[1] N. Burgess. "Spatial Cognition and the Brain". In: *Annals of the New York Academy of Sciences* 1124.1 (2008), pp. 77–97. DOI: 10.1196/annals.1440.002.

[2] J. Wu, X. Zhang, H. Yuan, X. Zhang, T. Huang, C. He, et al. *Visual Generation Unlocks Human-Like Reasoning through Multimodal World Models*. 2026. arXiv: 2601.19834.

[3] K. Park, Y. J. Choe, and V. Veitch. "The Linear Representation Hypothesis and the Geometry of Large Language Models". *ICML* 2024.

[4] C. Min, J. Jung, D. Lee, H. Jeon, Y. Su, J. Tremblay, C. H. Song, and J. Park. "Why Far Looks Up: Probing Spatial Representation in Vision-Language Models". *ECCV* 2026.

[5] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. "Microsoft COCO: Common Objects in Context". *ECCV* 2014.

[6] S. Tong, E. Brown, P. Wu, S. Woo, M. Middepogu, S. C. Akula, et al. "Cambrian-1: A Fully Open, Vision-Centric Exploration of Multimodal LLMs". *NeurIPS* 2024.

[7] M. Du, B. Wu, Z. Li, X. Huang, and Z. Wei. "EmbSpatial-Bench: Benchmarking Spatial Understanding for Embodied Tasks with Large Vision-Language Models". *ACL* 2024.

[8] H. Fang, J. Duan, D. Clay, S. Wang, S. Liu, W. Huang, et al. *MolmoAct2: Action Reasoning Models for Real-world Deployment*. 2026. arXiv: 2605.02881.

[9] Qwen Team. *Qwen3.5: Towards Native Multimodal Agents*. Feb. 2026. URL: https://qwen.ai/blog?id=qwen3.5.

[10] M. Huh, B. Cheung, T. Wang, and P. Isola. "Position: The Platonic Representation Hypothesis". *ICML* 2024.

[11] S. Yang, R. Xu, Y. Xie, S. Yang, M. Li, J. Lin, et al. "MMSI-Bench: A Benchmark for Multi-Image Spatial Intelligence". *ICLR* 2026.

[12] Z. Wan, Y. Xie, C. Zhang, Z. Lin, Z. Wang, S. Stepputtis, D. Ramanan, and K. P. Sycara. "InstructPart: Task-Oriented Part Segmentation with Instruction Reasoning". *ACL* 2025.

[13] Y. Zhang, R. Corcodel, C. Hori, A. Cherian, and D. Zhao. "SpinBench: Perspective and Rotation as a Lens on Spatial Reasoning in VLMs". *ICLR* 2026.

[14] P. Rajasekaran. *Harness Design for Long-Running Application Development*. Anthropic. Mar. 24, 2026. URL: https://www.anthropic.com/engineering/harness-design-long-running-apps.

[15] L. Chen, W. Xie, Y. Liang, H. He, H. Zhao, Z. Yang, et al. "BabyVision: Visual Reasoning Beyond Language". *ICML* 2026.

[16] L. Sharkey, B. Chughtai, J. Batson, J. Lindsey, J. Wu, L. Bushnaq, et al. "Open Problems in Mechanistic Interpretability". In: *Transactions on Machine Learning Research* (2025). arXiv: 2501.16496.

[17] S. Cho, R. Hachiuma, A. Badki, H. Su, B.-K. Lee, C. H. Song, et al. *SpatialClaw: Rethinking Action Interface for Agentic Spatial Reasoning*. 2026. arXiv: 2606.13673.

[18] NVIDIA. *Cosmos 3: Omnimodal World Models for Physical AI*. 2026. arXiv: 2606.02800.

[19] W. Gurnee, N. Sofroniew, A. Pearce, M. Piotrowski, I. Kauvar, R. Chen, et al. "Verbalizable Representations Form a Global Workspace in Language Models". In: *Transformer Circuits Thread* (2026). URL: https://transformer-circuits.pub/2026/workspace/index.html.

[20] R. Kang, H. Chen, G. Gkioxari, and P. Perona. "Linear Mechanisms for Spatiotemporal Reasoning in Vision Language Models". *ICLR* 2026.

Appendix

A S-Space Algorithm

A.1 Notation

Throughout this section, we fix a particular layer ℓ and omit the layer index from all formulas. Each example consists of an image and a prompt p that queries the position of one object, referred to as the **query**, relative to another, referred to as the **reference**, along a spatial axis $g \in \{x, y, z\}$. For example:

"`Is the apple to the left or right of the cup? Answer with left or right.`"

Here, *apple* is the query object, *cup* is the reference object, and the prompt queries their relative position along the horizontal axis. Let q and r denote the final token positions of the query and reference object names, respectively. Let $z_p(w)$ denote the final-answer logit assigned to endpoint token w. For each spatial axis, we define a fixed answer contrast:

$$s_{p,x} = z_p(\text{right}) - z_p(\text{left}), \qquad s_{p,y} = z_p(\text{up}) - z_p(\text{below}), \qquad s_{p,z} = z_p(\text{close}) - z_p(\text{far}).$$

Following prior work [19], we define the Jacobian of the answer contrast with respect to the hidden state at an object-token position t as

$$d_{p,t,g} := \nabla_{h_t} s_{p,g} \in \mathbb{R}^{d_{\text{model}}}.$$

This Jacobian lies in the same representation space as h_t and specifies the direction in which perturbing the object representation would increase the corresponding answer contrast.

A.2 Assumptions

Following the linear representation hypothesis [3, 20], we assume that semantic information can be linearly read out from hidden representations. In this work, we focus on viewer-centered spatial information and model its linearly readable variation through the three-dimensional subspace

$$\mathcal{S} = \text{span}\{v_x, v_y, v_z\}.$$

Assumption 1 (Representational decomposition). At an object-token position t, the hidden state can be conceptually decomposed into three mutually orthogonal components:

$$h_t = h_t^{\mathcal{S}} + h_t^{\text{non-spatial}} + h_t^{\text{context}},$$

where

$$h_t^{\mathcal{S}} \in \mathcal{S}, \qquad h_t^{\text{non-spatial}}, h_t^{\text{context}} \in \mathcal{S}^{\perp}.$$

Here, $h_t^{\mathcal{S}}$ encodes the object's viewer-centered spatial attributes; $h_t^{\text{non-spatial}}$ captures its non-spatial attributes, such as identity, appearance, and function; and h_t^{context} captures information contributed by the question semantics, prompt phrasing, and broader scene context.

We analogously decompose the Jacobian into components along the same three mutually orthogonal subspaces:

$$d_{p,t,g} = d_{p,t,g}^{\mathcal{S}} + d_{p,t,g}^{\text{non-spatial}} + d_{p,t,g}^{\text{context}}.$$

Assumption 2 (Component behavior). For a relative spatial question, the three Jacobian components satisfy

$$d_{p,q,g}^{\mathcal{S}} = -d_{p,r,g}^{\mathcal{S}},$$

$$d^{\text{non-spatial}}_{p,q,g} = d^{\text{non-spatial}}_{p,r,g} = 0,$$

and

$$\mathbb{E}\left[d^{\text{context}}_{p,q,g} - d^{\text{context}}_{p,r,g}\right] = 0,$$

where the expectation is taken over the prompt and example distribution. The first condition states that the spatial contribution changes sign when the query and reference roles are exchanged. The second states that non-spatial object attributes do not affect the spatial answer. The third assumes that role-asymmetric contextual effects cancel in expectation.

A.3 Algorithm 1: Identifying S-Space

To connect visual relations to explicit semantic directions, we construct relative spatial question-answering examples from 6,000 object pairs in uncropped COCO images [5], with 2,000 pairs assigned to each spatial axis. Each pair has a clear horizontal, vertical, or depth ordering, and the two endpoints of the corresponding axis serve as the possible answers.

Our core idea is that, if a model forms an internal spatial representation space, its final answer to a spatial question should, on average, reflect information in $\mathcal{S}$ rather than $\mathcal{S}^{\perp}$.

For each prompt p and axis g, we compute the role-antisymmetrized Jacobian

$$a_{p,g} = \frac{1}{2}\left(d_{p,q,g} - d_{p,r,g}\right) = \frac{1}{2}\left(\nabla_{h_q} s_{p,g} - \nabla_{h_r} s_{p,g}\right).$$

This operation isolates the component that changes sign when the query and reference roles are exchanged.

Property. Under Assumptions 1 and 2, role antisymmetrization isolates the spatial Jacobian component in expectation:

$$\mathbb{E}[a_{p,g}] = \mathbb{E}\left[d^{\mathcal{S}}_{p,q,g}\right].$$

Indeed,

$$a_{p,g} = \frac{1}{2}\left(d^{\mathcal{S}}_{p,q,g} - d^{\mathcal{S}}_{p,r,g}\right) + \frac{1}{2}\left(d^{\text{context}}_{p,q,g} - d^{\text{context}}_{p,r,g}\right)$$

$$= d^{\mathcal{S}}_{p,q,g} + \frac{1}{2}\left(d^{\text{context}}_{p,q,g} - d^{\text{context}}_{p,r,g}\right),$$

where the non-spatial terms vanish and the spatial terms combine by role antisymmetry. Taking expectations eliminates the remaining contextual difference, yielding the stated result.

To estimate this expectation empirically, we construct five prompt variants for each object pair and evaluate each variant with the two objects appearing in both possible orders. For each axis g, let $\mathcal{D}_g$ denote its 2,000 object-pair examples and let $\mathcal{P}_g$ denote the ten prompt configurations formed by the five templates and two object orderings. Using the superscript i to index examples, we compute

$$\bar{a}_g = \frac{1}{|\mathcal{D}_g|\,|\mathcal{P}_g|} \sum_{\substack{i \in \mathcal{D}_g \\ p \in \mathcal{P}_g}} a^{(i)}_{p,g}, \qquad v_g = \frac{\bar{a}_g}{\|\bar{a}_g\|_2}.$$

The resulting unit vectors v_x, v_y, and v_z define the horizontal, vertical, and depth axes of S-Space, respectively. Their positive directions correspond to right, up, and close.

Implementation details. We use the following five templates for the horizontal axis, replacing the endpoint terms as appropriate for the vertical and depth axes:

1. "`Is the {query} to the left or right of the {reference}? Answer with left or right.`"

2. "`Where is the {query} relative to the {reference}: left or right? Answer with left or right.`"
3. "`In the image, is the {query} to the left or right of the {reference}? Answer with left or right.`"
4. "`Which describes the position of the {query} relative to the {reference}: left or right? Answer with left or right.`"
5. "`Relative to the {reference}, is the {query} on the left or right? Answer with left or right.`"

A.4 Algorithm 2: Readout with S-Space

Once the three axes have been identified, reading out S-Space requires only a forward pass. Importantly, this readout is not restricted to the spatial QA prompts used for identification: once learned, the same directions can be applied to object-token representations elicited by other prompts and tasks. Stacking the learned directions as

$$V = [v_x, v_y, v_z] \in \mathbb{R}^{d_{\text{model}} \times 3},$$

we assign each object a model-native S-Space coordinate using the hidden state at the final token t of its name:

$$\mathbf{c}_t = V^\top h_t = V^\top h_t^{\mathcal{S}} \in \mathbb{R}^3.$$

The second equality follows because the columns of V lie in $\mathcal{S}$, whereas the non-spatial and contextual components lie in $\mathcal{S}^\perp$. Increasing coordinate values correspond to right, up, and close along the horizontal, vertical, and depth axes; decreasing values correspond to left, below, and far.

For a relative spatial question, we compare the query and reference readouts along the queried axis:

$$m_{q,r,g} = v_g^\top (h_q - h_r) = c_{q,g} - c_{r,g}.$$

The sign of $m_{q,r,g}$ directly selects the predicted endpoint: a positive value indicates right, up, or close, whereas a negative value indicates left, below, or far.

B S-Space Across Different Layers

To assess the sensitivity of S-Space to layer selection, we independently construct an S-Space readout at each layer of every model and evaluate its accuracy on spatial-relation questions. As shown in Appendix Figure B1, S-Space does not depend on a narrowly tuned layer choice: across all models, accuracy remains similarly high over a contiguous range of layers near the peak in the middle of the model's workspace. Performance drops primarily near the beginning and end of the workspace, indicating that S-Space is robust to the precise layer choice as long as an intermediate layer is used.

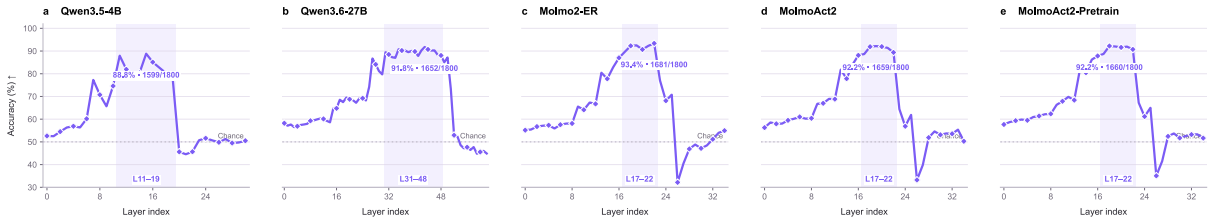

Figure B1 Layer-wise S-Space readout accuracy across models. Layer-wise S-Space readout accuracy on the 1,800-example COCO validation set. Shaded bands mark each model's high-accuracy layer range.