WHERE DO SPATIAL ANSWERS COME FROM? TRACING UTILITY ACROSS THE VISUAL SCENE

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-language models (VLMs) have made rapid progress in video-based spatial reasoning. However, it remains unclear which visual features these models rely on to generate their answers. Attention maps and representation probes reveal where a model attends and what information it encodes, but do not by themselves establish which feature directions a particular answer depends on. We introduce Utility Probe (U-probe), a training-free method that identifies an answer-conditioned feature subspace, termed Utility Space (U-Space), by coupling visual representations with gradients of the answer. Deletion and steering interventions validate the causal role of U-Space, whose contributions we trace across the visual scene. The leading feature directions in U-Space are interpretable as spatial concepts, and the subspaces for corresponding tasks agree across datasets. Deployed on VLMs finetuned on spatial tasks, U-probe reveals that scene-level spatial reasoning is supported by not only the queried objects but also features distributed over the scene, particularly on walls and floors, i.e., ***scene structure***. During finetuning, regional utility after as few as 20 steps predicts later model performance, providing an early diagnostic of spatial SFT, whereas regional attention does not. Our results connect answer-conditioned feature subspaces to their use across the scene and identify scene structure as an important substrate for spatial reasoning.

1 INTRODUCTION

Vision-language models (VLMs) (Bai et al., 2025a;b; Hong et al., 2025; Wang et al., 2025; Qwen, 2026) have become competent at spatial reasoning from images and videos, including estimating distances and sizes, counting objects, and reasoning about relative directions in complex scenes (Bai et al., 2025a; Lin et al., 2025; Yang et al., 2026b;a; Wu et al., 2026b; Zhang et al., 2026c). Supervised finetuning (SFT) on spatial data (Yang et al., 2026a;b; Zhang et al., 2026a) further improves these capabilities. However, a basic question remains unanswered:

What visual information does a VLM actually use to produce a spatial answer?

Answering this question requires distinguishing what a model represents from what it actually uses. We study this distinction at the level of answer-supporting feature subspaces. A visual token may encode several properties of the same region, e.g., both appearance and geometry. Attending to that token does not reveal which property supports a spatial answer. Attention-based analyses reveal attended or influential components (Chen et al., 2025; Ma et al., 2026; Gandikota & Bau, 2026; Chowdhury et al., 2026), while feature analyses reveal what spatial information is encoded in hidden representations (Wang & Gao, 2026; Cui et al., 2026; Min et al., 2026; Gopinath et al., 2026). These observations alone leave the answer's dependence on that information unresolved (Wang & Xu, 2026; Liao et al., 2026).

To bridge this gap, we introduce Utility Probe (U-probe), a training-free method for identifying Utility Space (U-Space), a compact, answer-conditioned visual feature subspace. U-probe couples token representations with gradients of the answer score, yielding directions along which represented content and answer sensitivity coincide. We validate their causal role by intervention: removing U-Space features reduces answer confidence, and moving representations along U-Space feature directions systematically steers generated spatial answers. The resulting directions are interpretable as spatial concepts and show cross-dataset agreement for corresponding spatial tasks.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

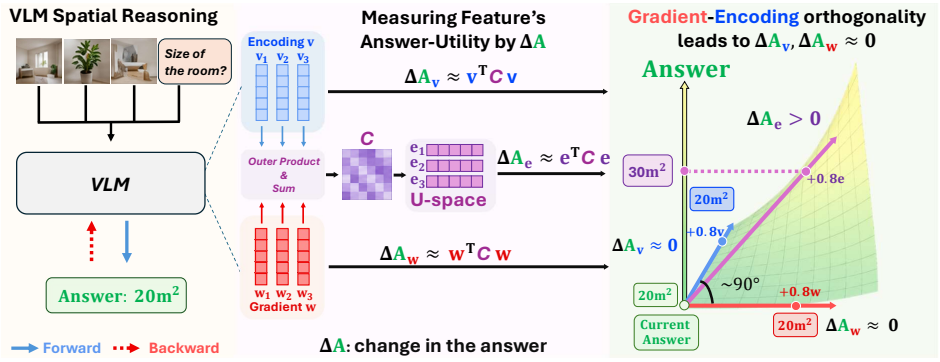


Figure 1: **What features does a VLM actually use for spatial reasoning?** Utility Probe (U-probe) couples encoded features and answer gradient, identifying the answer-relevant feature directions as Utility Space (U-Space).

Having identified which feature directions support an answer, we next ask where their contributions lie in the visual scene. We quantify each token's contribution within U-Space as its utility and trace it to semantic regions of the input. Across five VLMs on VSI590K/VSI-Bench (Yang et al., 2025; 2026b) and a video set constructed from SPAR-7M (Zhang et al., 2026a), scene-demanding answers run on not only the queried objects but also features distributed across the scene, particularly walls and floors, which we term `scene structure`. Feature interventions confirm that `scene structure` regions carry causally relevant features for questions about the room and spatial relations within it, even when they are not explicitly mentioned in the question.

We also find that SFT largely redirects the use of visual content already present in the base model. During SFT, stronger models retain a larger fraction of utility magnitude on scene structure for scene-demanding questions. Using only training samples without ground-truth labels, regional utility after as few as 20 SFT steps predicts the models' final accuracy gain from the visual input on held-out evaluation, whereas the corresponding regional attention shows no significant correlation. This provides an early diagnostic of spatial finetuning.

Our results establish a connection from feature directions, to causal dependence of the answer, to semantic regions of the scene, and finally to model performance. In summary, our contributions are:

- **U-probe and U-Space:** an answer-conditioned analysis yielding interpretable, causally validated feature directions that agree across datasets for corresponding spatial tasks.
- **Spatial reasoning runs on scene structure:** we show that scene structure regions, i.e., walls and floors, carry causally relevant features for spatial reasoning. In scene-demanding questions, it carries answer-relevant features in addition to those on the queried objects.
- **An early diagnostic of spatial SFT:** regional utility predicts final vision margins across models from as few as 20 SFT steps.

2 Related Work

Spatial reasoning and its mechanisms in VLMs. Recent VLMs have made rapid progress in spatial reasoning on images and videos, including object localization, distance and size estimation, relative direction, and scene-level understanding (Yang et al., 2025; Lin et al., 2025; Yang et al., 2026b;a; Zhang et al., 2026a; Wu et al., 2026b; Zhang et al., 2026c). Alongside improving model performance, a growing body of work investigates how such reasoning is implemented inside VLMs. One line analyzes how visual information is read or attended, through visual attention, specialized attention heads, or visual cues (Chen et al., 2025; Gandikota & Bau, 2026; Ma et al., 2026; Chowdhury et al., 2026; Wang et al., 2026). These studies reveal important functions of the model's visual attention, while others show that attention and answer correctness can also dissociate (Liu et al., 2026a). A complementary line probes hidden representations and finds that VLMs encode spatial variables such as scene topology, spatial relations, and object orientation (Wang & Gao, 2026; Cui et al., 2026; Min et al., 2026; Gopinath et al., 2026). However, information being encoded does not necessarily imply that it is used by the generated answer (Wang & Xu, 2026; Liao et al., 2026). Our U-probe connects these two views by identifying answer-conditioned directions within token

representations that the generated answer depends on, and tracing their contributions to semantic regions of the scene.

Attribution and relevant feature subspaces. Gradient-based attribution methods relate model internal representations to its outputs (Shrikumar et al., 2017; Selvaraju et al., 2017; Sundararajan et al., 2017), and recent work has adopted such signals to characterize spatial evidence in VLMs (Tan et al., 2026). Most closely related to our mathematical construction, Principal Relevant Component Analysis (PRCA) (Chormai et al., 2024) identifies low-dimensional activation subspaces that concentrate prediction relevance. U-probe instead derives answer-conditioned feature directions from the first-order effect of feature deletion, validates them by intervention, and traces their causal contributions across semantic regions of the visual scene.

3 Utility Probe and Utility Space

3.1 Preliminaries and notations

Vision-language models (VLMs) process image tokens $\mathcal{P}$ from its vision encoder and language tokens $\mathcal{T}$ jointly in a large language model (LLM). Denote $P = |\mathcal{P}|$, $T = |\mathcal{T}|$ for their number. For a token at position p in layer l of the LLM, let $\mathbf{h}_p^l \in \mathbb{R}^d$ denote its feature. The LLM answers the question by generating language tokens. Denote F as the log probability the model assigns to the answer $a = (a_1, \ldots, a_T)$, where $F(a) = \sum_{t=1}^{T} \log p\big(a_t \,\big|\, \mathcal{P}, \mathcal{T}, a_{<t}\big)$. The answer tokens can either come from the model's own generation a_{own}, or from the ground-truth (GT) a_{gt}. Accordingly, F can be instantiated as F_{own} and F_{gt}. The gradient of F w.r.t. $\mathbf{h}_p^l$ is $\mathbf{g}_p^l = \partial F / \partial \mathbf{h}_p^l \in \mathbb{R}^d$. The pair $(\mathbf{h}_p^l, \mathbf{g}_p^l)$ is all that U-probe uses; nothing requires training.

3.2 Procedure of U-probe and definition of U-Space

U-probe asks which feature directions are actually used by the answer. Two conditions must hold for a direction $\mathbf{d}$: the token must contain content along $\mathbf{d}$, measured by $\mathbf{d}^\top \mathbf{h}_p^l$, and the answer must be sensitive to perturbations along the same direction, measured by $\mathbf{d}^\top \mathbf{g}_p^l$. The product, $(\mathbf{d}^\top \mathbf{h}_p^l)(\mathbf{d}^\top \mathbf{g}_p^l) = \mathbf{d}^\top \big(\mathbf{h}_p^l \mathbf{g}_p^{l\top}\big)\mathbf{d}$, therefore quantifies the coupling between what token p carries and what the answer reads along $\mathbf{d}$. This motivates the outer product $\mathbf{h}_p^l \mathbf{g}_p^{l\top}$, that represents how token content co-occurs with answer sensitivity across all pairs of feature dimensions. Averaging these token-level couplings over image tokens gives the coupling matrix in Eq. 1:

$$\mathbf{C}^l = \frac{1}{P} \sum_{p \in \mathcal{P}} \mathbf{h}_p^l \, \mathbf{g}_p^{l\top} \in \mathbb{R}^{d \times d}. \tag{1}$$

We drop the superscript l from here on and will state it when it matters. $\mathbf{C}$ is asymmetric in that its left and right factors play different roles: the left factor represents the content carried by the token, while the right factor represents the direction along which the answer reads that content. To identify the dominant couplings between these two roles, we first consider the singular value decomposition (SVD), $\mathbf{C} = \sum_i \sigma_i \mathbf{v}_i \mathbf{w}_i^\top$, where each singular value σ_i measures the strength of the coupling between a content direction $\mathbf{v}_i$ and a read direction $\mathbf{w}_i$. Empirically, the paired directions are nearly orthogonal across VLMs (Appendix B), suggesting that neither alone provides a direct causal handle on the answer. To see the reason, consider removing a unit direction $\mathbf{d}$ from every image token, $\mathbf{h}_p \leftarrow \mathbf{h}_p - (\mathbf{d}^\top \mathbf{h}_p)\mathbf{d}$. This intervention changes the answer score to first order by:

$$\Delta F = -\sum_{p \in \mathcal{P}} (\mathbf{d}^\top \mathbf{h}_p)(\mathbf{d}^\top \mathbf{g}_p) = -P \, \mathbf{d}^\top \mathbf{C} \, \mathbf{d}, \tag{2}$$

Therefore, choosing either singular vector as the intervention direction gives $\mathbf{v}^\top(\sigma \mathbf{v}\mathbf{w}^\top)\mathbf{v} = \mathbf{w}^\top(\sigma \mathbf{v}\mathbf{w}^\top)\mathbf{w} = \sigma \mathbf{v}^\top \mathbf{w}$, which is approximately 0 given that the paired $\mathbf{v}$ and $\mathbf{w}$ are near-orthogonal. The same holds for the full $\mathbf{C}$, since the singular vectors on each side are orthonormal: $\mathbf{v}_i^\top \mathbf{C} \mathbf{v}_i = \mathbf{w}_i^\top \mathbf{C} \mathbf{w}_i = \sigma_i \mathbf{v}_i^\top \mathbf{w}_i$. To obtain meaningful intervention directions, we note that the quadratic form in Eq. 2 depends only on the symmetric part of $\mathbf{C}$:

$$\Delta F = -P \, \mathbf{d}^\top \mathbf{C} \, \mathbf{d} = -P \, \mathbf{d}^\top \mathbf{S} \, \mathbf{d}, \qquad \mathbf{S} = \tfrac{1}{2}\big(\mathbf{C} + \mathbf{C}^\top\big), \tag{3}$$

The eigenvectors of $\mathbf{S}$ are the desired intervention directions. The reason can be seen through the single coupling $\sigma \mathbf{v}\mathbf{w}^\top$. For each singular vector pair $(\mathbf{v}, \mathbf{w})$, the symmetric part of its rank-one

coupling $\sigma \mathbf{v}\mathbf{w}^\top$ has one normalized bisector $\mathbf{b} = (\mathbf{v} + \mathbf{w})/\sqrt{2(1 + \mathbf{v}^\top \mathbf{w})}$ as an eigenvector, with intervention effect $\mathbf{b}^\top(\sigma \mathbf{v}\mathbf{w}^\top)\mathbf{b} = \frac{\sigma}{2}(1 + \mathbf{v}^\top \mathbf{w})$, a non-zero number even when $\mathbf{v}^\top \mathbf{w} \approx 0$, given $\sigma > 0$. For the full $\mathbf{S}$, the intervention effect along any unit direction $\mathbf{d}$ sums the contributions from all singular vector pairs, $\mathbf{d}^\top \mathbf{S}\mathbf{d} = \sum_i \sigma_i (\mathbf{d}^\top \mathbf{v}_i)(\mathbf{d}^\top \mathbf{w}_i)$. The eigenvectors of $\mathbf{S}$ are exactly the directions that extremize this aggregated effect. Our construction is algebraically related to Principal Relevant Component Analysis (PRCA) (Chormai et al., 2024). When PRCA is instantiated with Gradient×Input, its context vector is the output gradient and its symmetrized activation-context matrix coincides, up to a constant factor, with $\mathbf{S}$ when the expectation is taken over token positions. The distinction is therefore not the eigendecomposition in itself, but the object it is used to characterize. PRCA derives a relevant subspace as a dataset-level decomposition of attribution, whereas U-probe couples image token representations with answer gradients in a VLM inference, and derives its eigendirections directly from the first-order effect of intervening on the hidden representation. Consequently, U-Space is used not only to explain where relevance lies, but as an intervention space whose directions can be removed from or injected into the representation to test necessity and steer the generated answer.

We define U-Space from $\mathbf{S}$. Write $\mathbf{S} = \sum_j \lambda_j \mathbf{e}_j \mathbf{e}_j^\top$ with the positive eigenvalues ordered by magnitude, $\lambda_1 \geq \lambda_2 \geq \dots$. The rank-$K$ U-Space is:

$$\mathcal{U}_K = \mathrm{span}\{\mathbf{e}_1, \dots, \mathbf{e}_K\}, \tag{4}$$

Unlike the two separate singular vector sets of $\mathbf{C}$, U-Space provides a single set of feature directions that can be directly removed from or added to token representations, making them explicit causal handles on the model's answer. $\mathcal{U}_K$ overlaps only partially with the principal directions of the token features and with those of the gradients. It is neither the dominant encoding directions nor the leading read directions. At $K = 4$, the mean principal-angle cosine is 0.32 on average across models and samples with the top-4 encoding directions, and 0.36 with the gradient ones. Probing either side alone does not fully recover it.

Deriving vision U-Space in the language part. Language tokens read the image tokens through attention, so a language token feature carries vision-derived content, including content that no single image token carries, such as the relation between two objects. To isolate that content approximately, we run the same input once more with the image tokens removed and take the difference, $\Delta \mathbf{h}_t = \mathbf{h}_t - \mathbf{h}_{t,\varnothing}$ for a language token $t \in \mathcal{T}$. The construction of Eq. (1)–(4) then applies unchanged with $(\Delta \mathbf{h}_t, \mathbf{g}_t)$ in place of $(\mathbf{h}_p, \mathbf{g}_p)$, where $\mathbf{g}_t = \partial F / \partial \mathbf{h}_t$, and yields the language-side U-Space. This U-Space records how the model composes what it has read from vision, which is what a relational answer such as left versus right requires.

Token utility. U-Space allows us to quantify how much answer-relevant content each token contributes. For an eigenvector $\mathbf{e}_j$ of $\mathbf{S}$, the token carries content along this direction by $\mathbf{e}_j^\top \mathbf{h}_p$, while the answer reads along it by $\mathbf{e}_j^\top \mathbf{g}_p$. Their product therefore measures the token's contribution through that utility direction. Summing over the K leading directions defines the per-token utility:

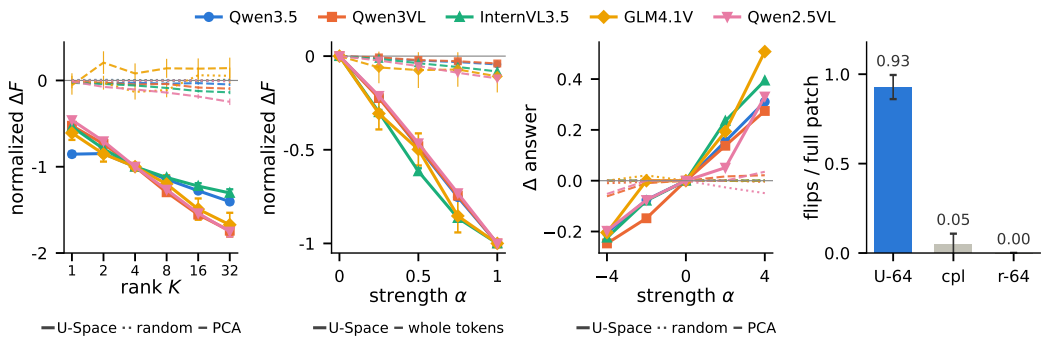
$$u_p^K = \sum_{j \leq K} (\mathbf{e}_j^\top \mathbf{h}_p)(\mathbf{e}_j^\top \mathbf{g}_p) = \mathbf{h}_p^\top \mathbf{P}_K \mathbf{g}_p, \qquad \mathbf{P}_K = \sum_{j \leq K} \mathbf{e}_j \mathbf{e}_j^\top. \tag{5}$$

Utility measures the alignment between what token p carries and what the answer reads within U-Space. Note that including all eigenvectors recovers the attribution method $u_p = \mathbf{g}_p^\top \mathbf{h}_p$ (Shrikumar et al., 2017; Selvaraju et al., 2017; Chormai et al., 2024; Tan et al., 2026).

We perform U-probe on 5 VLMs with similar size, Qwen3.5-9B (Qwen, 2026), InternVL3.5-8B (Wang et al., 2025), Qwen3VL-8B (Bai et al., 2025a), GLM4.1V-9B (Hong et al., 2025), Qwen2.5VL-7B (Bai et al., 2025b). We use two spatial datasets for SFT, VSI590K (Yang et al., 2026b) and a video training set that we build from SPAR-7M (Zhang et al., 2026a). During the SFT, the vision encoder is fixed and all other parameters are trainable. Details about the models and datasets are in Appendix A.

3.3 Causality of U-Space

U-Space is now a prediction of what the answer depends on. We test the prediction by intervention from two sides: deleting U-Space from the image tokens and measuring the answer score tests dependency, and moving the tokens along its axes and reading the generated answer tests steering.



(a) (b) (c) (d)

Figure 2: Causality of U-Space: dependency (a, b) and steering (c, d). **(a)**: The effect of deleting the positive span of the rank-K U-Space, the principle components, and random directions. **(b)**: The effect of removing a fraction α of the rank-4 U-Space and scaling the tokens. Normalized ΔF: the change of the answer score divided per model by the magnitude of its full $K = 4$ deletion. **(c)**: The output answer change after injecting the U-Space features, the principle components, and random directions to image tokens. **(d)**: Left/right answers that flip when the question tokens are patched by U-Space (U-64), its complement (cpl), or a random subspace. U-Space carries most of the answer-changing effects. Error bars are standard deviations, over questions in (a, b) and over models in (d).

Dependency. Recall that $\mathbf{P}_K$ is the projector onto $\mathcal{U}_K$. The deletion removes the U-Space component of every image token, $\mathbf{h}_p \leftarrow (\mathbf{I} - \mathbf{P}_K)\,\mathbf{h}_p$ for all $p \in \mathcal{P}$. The effect is measured by the change of the answer score, $\Delta F = F_{gt}^{\text{del}} - F_{gt}$ in nats. Deleting the negative span simply has the opposite effect, We compare the deletion with a random K-dimensional subspace R_K and the top-K principal components of the image token features, in mid-to-late layers in each VLM. We randomly choose 50 questions on VSI-bench and report the mean and the standard deviation (SD). Figure 2a shows the result. Deleting U-Space leads to much larger effects than the other two, indicating that the effect is neither because of a removal of dimensions or variances.

We further show that the dependency result cannot be reproduced by perturbing only the tokens most aligned with U-Space. Figure 2b compares two perturbations. The first removes a fraction α of the top-4 U-Space component from every image token, $\mathbf{h}_p \leftarrow \mathbf{h}_p - \alpha\,\mathbf{P}_4\mathbf{h}_p$. The second shrinks whole tokens, $\mathbf{h}_p \leftarrow (1-\beta)\,\mathbf{h}_p$, on the 10% of image tokens with the largest projection onto U-Space, with β chosen per item so that $\sum_p \|\Delta\mathbf{h}_p\|^2$ matches the first. We put the scaling after any normalization to ensure that the scaling is effective. The answer score falls with α along U-Space, whereas the token perturbation of the same budget barely moves it. Therefore, the effect is not explained simply by perturbing a small set of tokens most aligned with U-Space.

Steering. Deletion shows that the answer reads U-Space; steering asks whether moving the image tokens along its axes moves the generated answer. We choose 50 "room size" questions, a numeric type of VSI-Bench whose answer is continuous and not dominated by a prior (Wu et al., 2026b). We select the leading axis of the U-Space from the "room size" group, denoted as $\mathbf{u}$. During the forward pass, we add it to all image tokens at one layer: $\mathbf{h}_p \leftarrow \mathbf{h}_p + \alpha\, r\, (\mathbf{u}^\top \mathbf{h}_p)\mathbf{u}$, where r is the root mean square (RMS) of the image token features at that layer. Δ answer in Figure 2c is the median over the "room size" samples of the relative change of the generated answer, $(a_\alpha - a_0)/|a_0|$, with a_0 the original answer. The generated room size follows α monotonically in every model.

We also investigate the language-side vision U-Space, on relational direction questions. We focus on left/right answers and select 60 scenes, each with a pair of questions that name the same objects but with the opposite order, so that the correct answer flips accordingly. Let $\mathbf{P}_K^{lang}$ be the projector onto the rank-K with K=64 language-side U-Space, fitted on half of the scenes with the same left/right answer. On the other half, we run the first question in the pair and patch its question tokens at one layer with the features of the second, restricted to the subspace, $\mathbf{h}_t \leftarrow \mathbf{h}_t + \mathbf{P}_K^{lang}\,(\mathbf{h}_t^{swap} - \mathbf{h}_t)$ for $t \in \mathcal{T}$, and read the generated answer. The halves are exchanged and two runs are averaged. We count the flip of the left/right answer after the patching, and divide the number of answer flips by the one from full-feature patching where $\mathbf{P}_K^{lang} = \mathbf{I}$, i.e. the ceiling case. We compare the U-Space projection with its complement $\mathbf{I} - \mathbf{P}_K^{lang}$, and a random subspace of the same rank. The results are

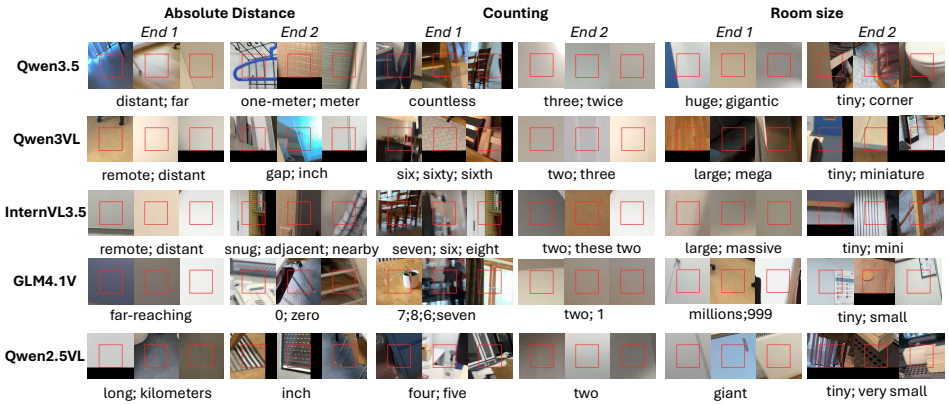


Figure 3: Interpreting the top U-Space axis of three question types. The image patches are the most extreme projections (End 1 and 2) onto the vision-side axis, where red box is the patch, shown with its surroundings. The logit lens of the language-side axis is shown under the patches.

in Figure 2d, averaged over 5 VLMs. U-Space-based patching steers the answer while the other two cannot. We report left/right here because front/back has a strong dataset prior and exhibits a distinct fixed-axis mechanism. See Appendix D.

3.4 Interpretability of U-Space in Spatial Reasoning

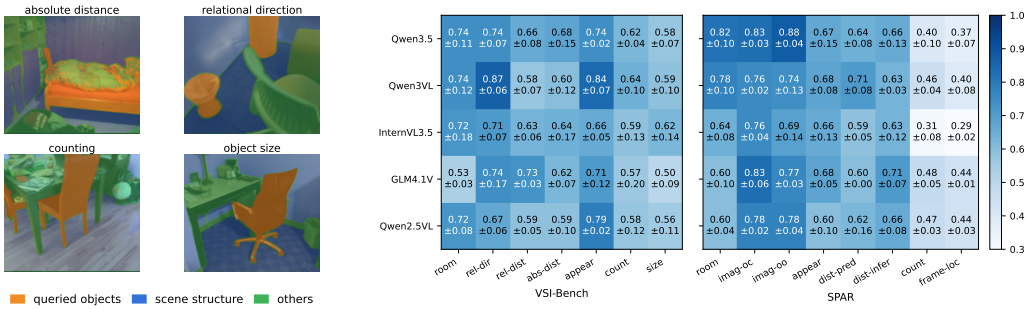
We demonstrate the interpretability of U-Space by showing what its leading axis corresponds to. For each question type, we average $\mathbf{S}$ over 50 questions of that type, build the U-Space, and read its top axis in two ways: by the image patches whose features project most extremely onto it, and by the logit lens (nostalgebraist, 2020) applied to the U-Space built from the language side of the same type. The two ends in logit lens of the language side are controlled by the sign of the axis. The logit lens assumes that the intermediate layer already has similar language features as the output layer, while the real alignment can be only partial (Belrose et al., 2023). Here, we keep our training-free framework. The aim is a human-readable interface on U-Space, not a complete lexicalization of it. Figure 3 shows the results. We translate all logit lens into English so some can be more than one token. The image patches show coherent visual concepts. For "Absolute Distance" and "Room Size", one end of the axis shows broad uniform surfaces or blank walls, and the other end collects the end of the plane, e.g., wall seams, corners, and edges. For "Counting", the two ends separate the objects one counts from the bare surfaces that contains nothing. The logit lens deliver human-interpretable description of the concepts. More results are in Appendix E. We note that, across question types and models, many patches highlight walls and floors areas instead of the queried objects, even if few question mentions those areas. Several similar visualizations and logit lens indicate that the different question types can share the use of similar features. We observe that the top axis in the "Absolute Distance" and the "Room Size" has a pair-wise cosine similarity within $[0.58, 0.72]$ across 5 VLMs, a quantitative evidence indicating that different spatial questions can depend on similar visual knowledge.

3.5 Cross-Dataset Consensus of U-Space

We test whether U-Space is specific to VSI590K or it generalizes across datasets. We repeat U-probe on the SPAR video set, whose questions come from different templates and sources. We compare the rank-K U-Space similarity using principal angle cosines, given by the singular values of $\mathcal{U}_{1,K}^{\top}\mathcal{U}_{2,K}$. For each question type on VSI, we find its partner type on SPAR by selecting the one with the highest U-Space similarity. Between partner types, the U-Space similarity is within $[0.68, 0.83]$, indicating a cross-dataset consensus of U-Space. The partner types are intuitive, e.g., "relational direction" in VSI finds "spatial imagination" in SPAR. Details of the results are in Appendix F.

3.6 SFT Reorganizes the Answer-Relevant U-Space

Our analysis runs on finetuned models, so we ask how their U-Space relates to that of the corresponding base models. The rank-4 U-Space similarity between the finetuned and base models are within $[0.13, 0.45]$ in every question type, indicating large reorganization during SFT while retaining partial overlap. We next ask which part of the coupling accounts for this overlap. We discover



(a) (b)

Figure 4: (a) The three regions on one frame of four scenes: the `queried objects` (orange), the `scene structure` (blue) and `others` (green), from the SAM3 masks. (b) Scene structure is necessary wherever the question needs scene information. Each cell is the fraction of questions of a type on which deleting the type's rank-4 U-Space on the `scene structure` tokens lowers the answer score more than the same deletion on a size-matched random token set.

that the content direction $\mathbf{v}$ does not change drastically between base and finetuned models, with cosine similarity within $[0.61, 0.82]$, while the read one $\mathbf{w}$ rotates drastically to new directions, with cosine similarity within $[0.01, 0.29]$. Dominant content directions are much more stable than read directions during SFT, suggesting that SFT primarily reorganizes how existing visual content is read.

4 Spatial Reasoning Runs on Scene Structure

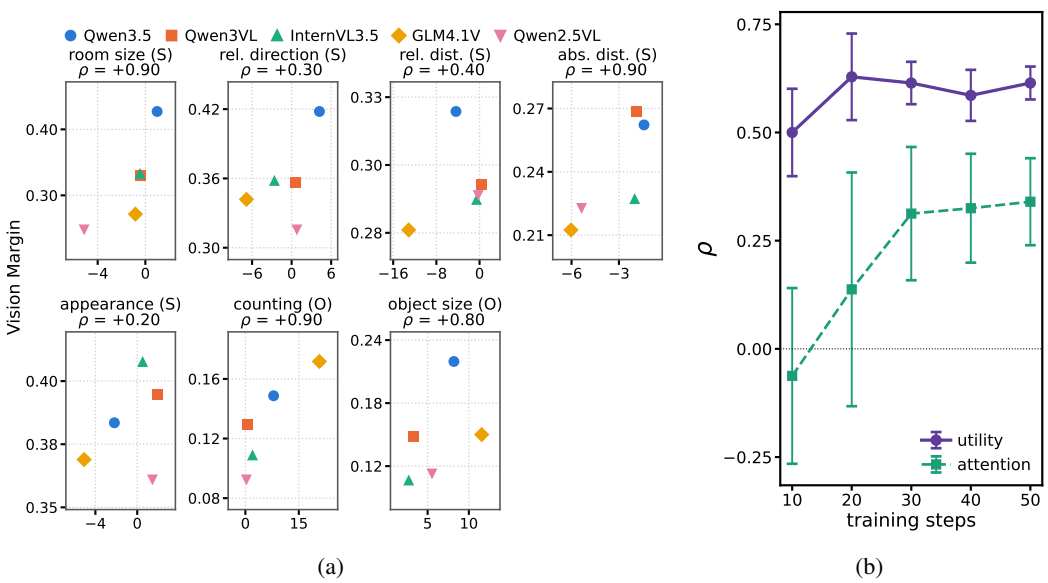
4.1 Where are U-Space Features Localized After SFT?

SFT reorganizes the answer-relevant U-Space, and Section 3.4 shows that U-Space features highlight non-object areas such as walls and floors. Prior work indicates qualitatively that the wall tokens carry features that matter for spatial reasoning (Wang et al., 2026). We therefore hypothesize that the seemingly unimportant areas (Chen et al., 2025; Liu et al., 2026a), e.g., walls and floors, play a critical role in spatial reasoning, and that U-Space features after SFT are localized on those areas, which we term scene structure.

Definition of scene structure. On every sampled video frame, we run SAM3 (Carion et al., 2026) with the text `wall` and `floor`, and take the union of the returned masks as the `scene structure`. The objects named in the question are segmented with their names, giving the `queried objects` region. An image token belongs to a region when at least half of its patch lies in the region's mask. Every token left is assigned to `others`. Examples are shown in Figure 4a. Averaged over scenes, the queried objects occupy roughly 10% of the image tokens, while scene structure and others are of similar size and split the remaining 90% nearly equally.

Comparison of importance between spatial regions. We test which region of this taxonomy matters for spatial reasoning. We adopt the deletion of Section 3.3, removing the rank-4 U-Space of the question type from the image tokens but restricted to one region, and compare its ΔF with the ΔF of the same-size deletion with randomly chosen tokens. A region matters within a question type when its deletion is more damaging than deleting a random set of size-matched tokens, on more than half of the questions. Note that U-Space remains the same across regions. We choose the layer to perform U-probe for each model, selecting mid-to-deep layer. We choose three disjoint 30-sample subsets for each question type and report the mean and standard deviation. Results of `scene structure` are in Figure 4b. `Scene structure` matters in many question types across 5 VLMs. `Queried objects` matters in all question types that involve objects, validating the correctness of our method. The result is in Appendix G. `Others` matters the least across the 5 models, containing irrelevant objects or background. Deleting the tokens themselves, however, cannot make this comparison. Zeroing tokens within a region is no more damaging than zeroing a random token set of the same size. We also compare the importance of `wall` with that of `floor`, and find that they matter roughly equally, with all cells in the damaging heatmap around 0.5.

We further calculate the density of the information carried by tokens per region, $\rho_\Omega = |\Delta F|/|\Omega|$ where Ω is one of the three regions and $|\Omega|$ denotes the number of tokens in that region. Per token, the `queried objects` is the densest. The conclusion is therefore not that `scene structure`



(a) (b)

Figure 5: The predictor on VSI-Bench. (a) X is the change of the region's fraction of $\sum_p |u_p|$ from the base model to step 20 of SFT, using the model's own generated answer. The region is scene structure (S) on the five scene types and the queried objects (O) on counting and object size. Y is the vision margin at step 800 on that type. ρ is the five-model Spearman correlation, on which we do not make any interpretation. (b) The aggregated ρ over all question types at each SFT step. The aggregation is tested under the null hypothesis that model labels are exchangeable, and the utility p-values are all under 0.02 while all attention ones are above.

carries more information per token than objects, but that a region covering $\sim$ 45% of the tokens, which no question names, carries causally relevant features wherever the question concerns the scene information. Moreover, we can roughly see a taxonomy of the question types. The `scene structure` region matters less in "count", "size", "frame loc" questions than the other ones, where those questions concern more on the object than the scene information.

4.2 Retaining the use of scene structure predicts better performance

Section 4.1 establishes where causally relevant features are localized, not how well a model uses it. Both `queried objects` and `scene structure` regions carry causally relevant features on many question types. We therefore ask: does the regional allocation of utility during SFT track how good it becomes? For a question type and a region Ω, we take the regional fraction of the answer's utility magnitude, $s_\Omega = \sum_{p\in\Omega} |u_p| / \sum_p |u_p|$, averaged over 100 training questions of the type on VSI590K. Let X be its change from the base model to SFT step W. Importantly, X needs no labels. The results are obtained under the model's own generated answer. Y is the vision margin of the model at SFT step 800 on that type, its accuracy with vision minus its accuracy without. We use vision margin rather than accuracy because utility measures image-derived information, and the margin isolates the accuracy attributable to vision. The 5 VLMs have similar accuracy without images, all within $[43.4, 44.0]$. The region selection is fixed in advance by a rule we develop. A scene-demanding type asks about the room or about where things stand in it, so it reads `scene structure`; "counting" and "object size" ask for a property of the named objects, so they read the `queried objects`. The predictor results are shown in Figure 5. Under our rule-based assignment of the region reading, the change of the regional utility tends to track the model performance. A seemingly counterintuitive phenomenon is that the `scene structure` utility fraction can decrease during SFT. This is because of the compositional nature of s_Ω and thus the competition between the `queried objects` and `scene structure`, as the model needs to use both for answering spatial questions. Even if in "room size", the models can lean towards recognizing the objects in the scene, implying a poor performance on estimating the room size. Therefore, it is retaining the usage of `scene structure` that tracks model performance on scene-demanding questions. Note that the utility change in `queried objects` and `others` shows a reverse trend against Y on those scene-demanding questions, as detailed in Appendix H.1.

Table 1: Accuracy (%) of the five VLMs finetuned on VSI590K for 800 steps, tested on VSI-Bench. †: The model is finetuned for a full epoch. **Bold** denotes the best, except the longer-trained Qwen3.5.

Model	Overall	Room	Abs. dist	Rel. dist	Rel. dir	Appear.	Count	Obj. size	Route
Qwen3.5	**71.7**	**75.2**	**57.9**	**73.1**	**88.5**	81.9	**74.2**	74.8	**48.5**
Qwen3VL	68.1	64.5	50.3	67.7	83.0	**84.0**	72.1	**74.9**	47.9
InternVL3.5	65.1	63.4	48.6	66.5	80.3	81.6	71.7	71.1	37.6
GLM4.1V	65.0	56.3	48.5	67.3	80.2	80.9	70.2	68.3	**48.5**
Qwen2.5VL	63.6	56.3	46.9	68.5	76.8	78.8	70.0	70.6	40.7
Qwen3.5†	73.1	78.7	58.7	73.7	89.3	83.7	74.3	77.7	48.5

With five models, a single type's rank correlation ρ is coarse, so we do not interpret its value. We aggregate per-type Spearman correlations and assess significance by exact permutation of the five model labels. Details are in Appendix H.2. Utility is significant at every checkpoint with $p \leq 0.017$, whereas the attention baseline is not in that $p \geq 0.09$. Attention is a common proxy for where visual information is read, yet it does not fully specify whether the model actually uses the attended content. VLMs can place large attention on tokens that carry nothing (Xiao et al., 2024; Kang et al., 2025). SPAR shows the same qualitative trend with an additional operation in region assignment. See Appendix H.3. Deleting the contribution from either `wall` or `floor` hurts the predictor performance, indicating both of them contribute to the predictive signal. Training loss disagrees with our predictor and can misplace models, or predicts the model performance without statistical significance, so does training set accuracy, see Appendix H.4. The predictor also successfully diagnoses a model it was not fitted on, see Appendix H.5.

4.3 Benchmark accuracy

We perform a 800-step SFT on VSI590K on the 5 VLMs, and evaluate them with RGB-only input on VSI-Bench. Details of the SFT are in Appendix A. The results are in Table 1. The best model, Qwen3.5, performs particularly well on scene-demanding questions such as "room size", "relational direction", and "absolute distance". Qwen3.5 is exactly the model that has the highest mean rank of X in our predictor, and our predictor spots its continuous usage of `scene structure` on those scene-demanding questions in Figure 5a. We thus train Qwen3.5 to a full epoch, reaching 73.1 overall accuracy on VSI-Bench, a new SOTA on VSI-Bench with only RGB and QA inputs during spatial SFT. See Appendix A.3 for the comparison with previous methods.

5 Discussions and limitations

U-probe identifies answer-relevant feature directions, but turning this diagnosis into improved accuracy remains open. U-Space perturbations move predictions causally, yet neither steering along an axis nor re-weighting attention onto `scene structure` raised benchmark accuracy; see Appendix I. Regularizing regional utility during SFT yields no clear gain in our setting, consistent with SFT changing read directions more strongly than content directions inherited from pretraining; see Appendix J. This suggests several directions for future work. Utility could guide pretraining toward features that preserve scene geometry before task-specific reading is learned. It could also motivate architectures with an explicit pathway for global scene structure, complementing the object-centric representations. As an early diagnostic, regional utility could further guide model, data, or hyperparameter selection before expensive SFT completes. Finally, extending the region taxonomy beyond indoor scenes could reveal the features that support spatial reasoning in open-world environments.

6 Conclusion

We introduce U-probe, a training-free method that couples VLM answer gradient with image token features to identify U-Space, an answer-relevant feature subspace. Feature interventions provide causal validation. Deleting U-Space changes the answer score more strongly than matched controls. Steering along it moves the generated answer. SFT reorganizes U-Space, with read directions changing much more than content directions. Across five VLMs and two video datasets, U-probe reveals that `scene structure` regions, i.e. walls and floors, carry causally relevant features alongside the queried objects. Across the five models, retaining more utility on `scene structure` during early SFT is positively correlated with larger final vision margins on scene-demanding questions, with the correlation visible after just 20 SFT steps. The model ranked the highest by this signal is also the strongest of the five after SFT, reaching 73.1 overall accuracy on VSI-Bench.

AI USE STATEMENT

In this work, we used generative AI tools for aiding paper writing and polishing the text. We also used them to write the code and execute the experiments, with human-checked validity and correctness. We have not used generative AI tools to generate synthetic data sets, help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, assist with translation, clean and reformat dataset, support qualitative and thematic data analysis, or interpret results. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work investigates existing vision-language models and public spatial-reasoning datasets, and does not involve human subjects or collection of personal data. We do not identify additional ethical concerns specific to this work.

REPRODUCIBILITY STATEMENT

We provide the training and evaluation details, model checkpoints, dataset information, experiment details in the Appendix. We will release the code upon acceptance.

REFERENCES

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025b.

Nora Belrose, Igor Ostrovsky, Lev McKinney, Zach Furman, Logan Smith, Danny Halawi, Stella Biderman, and Jacob Steinhardt. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*, 2023.

Zhongang Cai, Ruisi Wang, Chenyang Gu, Fanyi Pu, Junxiang Xu, Yubo Wang, Wanqi Yin, Zhitao Yang, Chen Wei, Tongxi Zhou, et al. Scaling spatial intelligence with multimodal foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7879–7890, 2026.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris Coll-Vinent, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. In *International Conference on Learning Representations*, volume 2026, pp. 138846–138923, 2026.

Shiqi Chen, Tongyao Zhu, Ruochen Zhou, Jinghan Zhang, Siyang Gao, Juan Carlos Niebles, Mor Geva, Junxian He, Jiajun Wu, and Manling Li. Why is spatial reasoning hard for vlms? an attention mechanism perspective on focus areas, 2025. URL https://arxiv.org/abs/2503.01773.

Pattarawat Chormai, Jan Herrmann, Klaus-Robert Müller, and Grégoire Montavon. Disentangled explanations of neural network predictions by finding relevant subspaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(11):7283–7299, 2024.

Rahul Chowdhury, Timothy A Rupprecht, Xuan Shen, Pu Zhao, and Yanzhi Wang. Scale: Attention head scaling as a minimal adapter for spatial reasoning in vision language models. *arXiv preprint arXiv:2606.29579*, 2026.

Kelly Cui, Nikhil Prakash, Shoval Messica, Ayush Raina, David Bau, Antonio Torralba, and Tamar Rott Shaham. The dual mechanisms of spatial variable binding in vision-language models. *arXiv preprint arXiv:2603.22278*, 2026.

Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, et al. Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 11198–11201, 2024.

Rohit Gandikota and David Bau. Gaze heads: How vlms look at what they describe. *arXiv preprint arXiv:2606.14703*, 2026.

Anju Gopinath, Nikhil Krishnaswamy, and Bruce Draper. Why mllms struggle to determine object orientations. *arXiv preprint arXiv:2604.13321*, 2026.

Bo Gu, Zhikang Zhang, Zizhuang Wei, Zhenyuan Chen, Lingyun Li, and Zhuoyi Song. Spacemind++: Toward allocentric cognitive maps for spatially grounded video mllms. *arXiv preprint arXiv:2605.09449*, 2026.

Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, Lihang Pan, et al. Glm-4.5 v and glm-4.1 v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. *arXiv preprint arXiv:2507.01006*, 2025.

Ting Huang, Zhenyu Zhang, Wenyuan Huang, Jian Yang, and Hao Tang. Consispace: Learning geometric consistency matters for video spatial reasoning. *arXiv preprint arXiv:2607.17599*, 2026.

Seil Kang, Jinyeong Kim, Junhyeok Kim, and Seong Jae Hwang. See what you are told: Visual attention sink in large multimodal models. In *International Conference on Learning Representations*, volume 2025, pp. 87676–87703, 2025.

Haoyuan Li, Qihang Cao, Tao Tang, Kun Xiang, Zihan Guo, Jianhua Han, JiaWang Bian, Hang Xu, and Xiaodan Liang. Thinking with geometry: Active geometry integration for spatial reasoning. *arXiv preprint arXiv:2602.06037*, 2026.

Chih-Ting Liao, Fei Shen, Xin Cao, and Tat-Seng Chua. Decodable is not grounded: A vision-ablation arbiter for vlm spatial reasoning. *arXiv preprint arXiv:2606.31257*, 2026.

Jingli Lin, Runsen Xu, Shaohao Zhu, Sihan Yang, Peizhou Cao, Yunlong Ran, Miao Hu, Chenming Zhu, Yiman Xie, Yilin Long, et al. Mmsi-video-bench: A holistic benchmark for video-based spatial intelligence. *arXiv preprint arXiv:2512.10863*, 2025.

Zhining Liu, Ziyi Chen, Hui Liu, Chen Luo, Xianfeng Tang, Suhang Wang, Jingying Zeng, Zhenwei Dai, Zhan Shi, Tianxin Wei, et al. Seeing but not believing: Probing the disconnect between visual attention and answer correctness in vlms. In *International Conference on Learning Representations*, volume 2026, pp. 72849–72876, 2026a.

Zuntao Liu, Yi Du, Taimeng Fu, Shaoshu Su, Cherie Ho, and Chen Wang. Vision-language memory for spatial reasoning. In *European Conference on Computer Vision*, pp. 297–316. Springer, 2026b.

Xueqi Ma, Shuo Yang, Yanbei Jiang, Shu Liu, Zhenzhen Liu, Jiayang Ao, Xingjun Ma, Sarah Monazam Erfani, and James Bailey. Attention in space: Functional roles of vlm heads for spatial reasoning. *arXiv preprint arXiv:2603.20662*, 2026.

Cheolhong Min, Jaeyun Jung, Daeun Lee, Hyeonseong Jeon, Yu Su, Jonathan Tremblay, Chan Hee Song, and Jaesik Park. Why far looks up: Probing spatial representation in vision-language models. *arXiv preprint arXiv:2605.30161*, 2026.

nostalgebraist. interpreting GPT: the logit lens, 2020. URL `https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens`. LessWrong blog post.

Team Qwen. Qwen3.5: Towards native multimodal agents, February 2026. URL `https://qwen.ai/blog?id=qwen3.5`.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. URL https://arxiv.org/abs/2408.00714.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 618–626, 2017.

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *International Conference on Machine Learning*, pp. 3145–3153. PMlR, 2017.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pp. 3319–3328. PMLR, 2017.

Kaizhen Tan, Yang Feng, and Heqing Du. Creg: Compass relational evidence graph for characterizing directional structure in vlm spatial-reasoning attribution. *arXiv preprint arXiv:2603.20475*, 2026.

Gemma Team, Sherif El Abd, Vaibhav Aggarwal, Robin Algayres, Alek Andreev, Olivier Bachem, Ian Ballantyne, Cormac Brick, Victor Cărbune, Michelle Casbon, et al. Gemma 4 technical report. *arXiv preprint arXiv:2607.02770*, 2026.

Haoming Wang and Wei Gao. Uncovering and shaping the latent representation of 3d scene topology in vision-language models. *arXiv preprint arXiv:2605.07148*, 2026.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Zeyu Wang and Xinming Xu. Knowing isn't always saying: When do spatial encodings reach answers in vision-language models? *arXiv preprint arXiv:2608.22916*, 2026.

Zhaozhi Wang, Tong Zhang, Mingyue Guo, Yaowei Wang, and Qixiang Ye. Videoanchor: Reinforcing subspace-structured visual cues for coherent visual-spatial reasoning. In *International Conference on Learning Representations*, volume 2026, pp. 11010–11028, 2026.

Jing Wu, Jianhua Wu, Jiayi Guan, Jiahong Chen, Jinghui Lu, Hangjun Ye, Bingzhao Gao, and Long Chen. Spatiolm: Towards general physical spatial intelligence in vision-language models. *arXiv preprint arXiv:2608.01899*, 2026a.

Mingrui Wu, Zhaozhi Wang, Fangjinhua Wang, Jiaolong Yang, Marc Pollefeys, and Tong Zhang. From indoor to open world: Revealing the spatial reasoning gap in mllms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16789–16799, 2026b.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *International Conference on Learning Representations*, volume 2024, pp. 21875–21895, 2024.

Jihan Yang, Shusheng Yang, Anjali W Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10632–10643. IEEE, 2025.

Jihan Yang, Zifan Zhao, Xichen Pan, Shusheng Yang, Junyi Zhang, Bingyi Kang, Hu Xu, Shang-Wen Li, and Saining Xie. Cambrian-p: Pose-grounded video understanding. *arXiv preprint arXiv:2605.22819*, 2026a.

Shusheng Yang, Jihan Yang, Pinzhi Huang, Ellis Brown, Zihao Yang, Yue Yu, Shengbang Tong, Zihan Zheng, Yifan Xu, Muhan Wang, et al. Cambrian-s: Towards spatial supersensing in video. In *International Conference on Learning Representations*, volume 2026, pp. 78185–78225, 2026b.

Jiahui Zhang, Yurui Chen, Yueming Xu, Ze Huang, Jilin Mei, Chunhui Chen, Yanpeng Zhou, Yu-Jie Yuan, Xinyue Cai, Guowei Huang, et al. From flatland to space: Teaching vision-language models to perceive and reason in 3d. *Advances in Neural Information Processing Systems*, 38, 2026a.

Jian Zhang, Shijie Zhou, Bangya Liu, Achuta Kadambi, and Zhiwen Fan. Spatialstack: Layered geometry-language fusion for 3d vlm spatial reasoning. *arXiv preprint arXiv:2603.27437*, 2026b.

Yiming Zhang, Jiacheng Chen, Jiaqi Tan, Yongsen Mao, Wenhu Chen, and Angel X Chang. Revsi: Rebuilding visual spatial intelligence evaluation for accurate assessment of vlm 3d reasoning. In *Forty-third International Conference on Machine Learning*, 2026c.

Ruosen Zhao, Zhikang Zhang, Jialei Xu, Jiahao Chang, Dong Chen, Lingyun Li, Weijian Sun, and Zizhuang Wei. Spacemind: Camera-guided modality fusion for spatial reasoning in vision-language models. *arXiv preprint arXiv:2511.23075*, 2025.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand, 2024. Association for Computational Linguistics. URL http://arxiv.org/abs/2403.13372.

A Experiment details

A.1 Models and datasets

Table 2 lists the five models. All models are obtained from the public checkpoints named in the table. Qwen3.5 is a hybrid: 24 of its 32 layers are linear-attention (Gated DeltaNet) layers and 8 are softmax-attention layers. We pick the softmax layers for analysis. For all analyses, we select a fixed relative depth across models, using the layer closest to 60% of the model depth.

Table 2: The five VLMs. Parameters from the checkpoints: total, vision encoder, projector (the merger modules, including Qwen3VL's DeepStack mergers) and language model.

	LM	Vision	Parameters (B)				Chosen
Checkpoint	Layers / width	Layers / width	total	vision	proj.	LM	layer
Qwen3.5-9B	32 / 4096	27 / 1152	9.41	0.42	0.04	8.95	19
Qwen3-VL-8B-Thinking	36 / 4096	27 / 1152	8.77	0.42	0.16	8.19	21
InternVL3.5-8B	36 / 4096	24 / 1024	8.53	0.30	0.03	8.19	21
GLM-4.1V-9B-Thinking	40 / 4096	24 / 1536	10.29	0.71	0.19	9.40	24
Qwen2.5-VL-7B-Instruct	28 / 3584	32 / 1280	8.29	0.63	0.04	7.62	17

Table 3 lists the datasets. VSI590K (Yang et al., 2026b) is the SFT set and VSI-Bench (Yang et al., 2025) its benchmark. The SPAR video set is built from SPAR-7M (Zhang et al., 2026a) with 32-frame videos and used both for SFT and, through its validation split, for evaluation. The SPAR training and validation sets share no scene. For the predictor analysis, we pair the training categories in VSI590K and the test ones in VSI-Bench: room size–room size, relative direction (object)–relative direction, relative distance (object)–relative distance, absolute distance–absolute distance, appearance order–appearance order, absolute count–object counting, absolute size (object)–object size. We sample the frame at 1 fps during training with maximum frame number 64, and test with uniformly sampled 64 frames on VSI, and the maximum and test frame number is 32 on SPAR.

A.2 Details of SFT

All models are finetuned with LlamaFactory (Zheng et al., 2024) with one recipe in Table 4: the vision encoder is frozen, the projector and the language model are trained. The learning-rate schedule spans one epoch of VSI590K (2308 steps at batch 256), and the models analyzed in the paper are the

Table 3: Details of the datasets we use.

Dataset	Size	Frames	Question types
VSI590K	590667 questions	up to 64 per video	relative direction (object, camera); relative distance (object, camera); absolute distance; relative size; absolute size; absolute direction; absolute count; relative count;appearance order; room size
VSI-Bench	5130 questions videos	64 sampled	object size; relational direction; absolute distance; relative distance; appearance order; counting; room size; route planning
SPAR video set	197540 training questions; 3200 validation questions	up to 32 per video	room size, spatial imagination (object-centric, object-to-object), appearance order, distance prediction, distance inference from a center point, object count, object frame location

Table 4: SFT configuration.

setting	value
optimizer	AdamW, $\beta = (0.9, 0.999)$, weight decay 0
learning rate	5×10^{-6}, linear decay to 0 over one epoch, no warmup
batch	256 (4 GPUs $\times$ 1 sample $\times$ 64 gradient-accumulation steps)
gradient clipping	1.0
image resolution	147456 to 262144 pixels per frame

checkpoints at step 800, about a third of the epoch. The runs on SPAR use the same recipe with 32 frames per video, also read at step 800. We run our all of our experiments on Nvidia GH200 GPUs.

Feature resize. 64 frames at the native resolution of the vision encoders induce huge computational overhead. We resize the frame after the vision encoder. The encoder runs at its native resolution on each frame, and its output feature map is bilinearly interpolated to at most 8×8 tokens, keeping the aspect ratio. For InternVL3.5, we resize the frames to 448×448 without tiling. The same resize is applied in training and in every evaluation and analysis.

Prompts. The three thinking models, Qwen3.5, Qwen3VL and GLM4.1V, are finetuned and evaluated with an empty thinking block in the prompt, so they answer directly; InternVL3.5 and Qwen2.5VL use their native chat templates. Evaluation uses VLMEvalKit (Duan et al., 2024).

A.3 Comparison with previous methods on VSI-Bench

Table 5 puts the full-epoch Qwen3.5 of Table 1 among the VSI-Bench results published for models that take RGB frames at test time, as reported in the respective papers, under the official protocol of Yang et al. (2025): numerical types are scored by mean relative accuracy, relational direction is the mean of its easy, medium and hard subsets, and the average is the mean of the eight task scores. Under the task mean, Qwen3.5 finetuned on VSI590K alone is the best model among those trained from RGB frames and question-answer pairs only; the models above it add camera-pose supervision (Yang et al., 2026a), a feed-forward geometry model on the frames (Gu et al., 2026), or both a geometry-consistent memory and reinforcement learning with consistency rewards (Huang et al., 2026), whose SFT model scores 71.2.

We can also see that Qwen3.5 falls behind the most significantly in Absolute Distance category, while it matches or surpasses the current SOTA model in other scene-demanding categories. We note that the answering indoor scene Absolute distance Questions could suffer from a prior, in that the distance between objects are within a small range in indoor scenes Wu et al. (2026b). For example, answer 1.5m for questions asking about the distance between a sink and a gas stove is nearly always true and can get a high score and low loss during SFT, while the model will be prone to ignoring vision in this case. This category also has the lowest vision margin among those scene-

Table 5: VSI-Bench accuracy (%) of different VLMs. Ours is the full-epoch Qwen3.5. Base: the language or vision-language model the method starts from. Extra: G, a feed-forward geometry model runs on the frames at test time; P, camera-pose supervision during training; D, depth supervision during training; RL, reinforcement learning after SFT. Column order as in Table 1. [a]: finetuned on VSI590K only, from Table 1 of Yang et al. (2026a).

Method	Base	Extra	Avg.	Room	Abs. dist	Rel. dist	Rel. dir	Appear.	Count	Obj. size	Route
Cambrian-S-7B (Yang et al., 2026b)	Qwen2.5-7B		67.5	72.2	50.5	71.1	76.2	80.1	73.2	74.9	41.8
Cambrian-S-7B[a] (Yang et al., 2026a)	Qwen2.5-7B		69.2	74.7	53.7	71.5	82.0	84.3	73.6	75.2	38.7
SenseNova-SI (Cai et al., 2026)	InternVL3-8B		68.8	72.8	53.5	69.6	80.8	76.4	72.0	76.8	48.5
Qwen3.5 (ours)[b]	Qwen3.5-9B		**73.1**	78.7	58.7	73.7	89.3	83.7	74.3	77.7	48.5
SpatioLM (Wu et al., 2026a)	SenseNova-SI	D	71.6	71.5	55.7	78.2	74.4	87.8	81.9	75.2	47.9
Cambrian-P (Yang et al., 2026a)	Qwen2.5-7B	P	73.7	76.9	60.1	74.8	89.5	85.0	74.9	76.0	52.6
SpatialStack-5B (Zhang et al., 2026b)	Qwen3.5-4B	G	67.5	68.2	55.6	67.3	84.1	83.5	71.0	69.1	41.2
VLM^2-7B (Liu et al., 2026b)	Qwen2-7B	G	68.8	69.9	59.6	69.0	87.8	68.3	72.5	70.8	52.6
SpaceMind (Zhao et al., 2025)	InternVL3-8B	G	69.6	74.2	61.4	67.2	88.4	70.6	73.3	77.3	44.3
GeoThinker (Li et al., 2026)	Qwen3-VL-8B	G	72.6	–	–	–	–	–	–	–	–
SpaceMind++ (Gu et al., 2026)	InternVL3-8B	G	73.2	76.9	62.4	73.5	89.7	84.1	74.5	77.9	48.2
ConsiSpace, SFT (Huang et al., 2026)	Qwen3-VL-8B	G	71.2	71.7	73.8	68.5	83.6	73.8	71.0	77.7	49.8
ConsiSpace, RL (Huang et al., 2026)	Qwen3-VL-8B	G, RL	**76.6**	76.3	78.9	72.1	90.1	79.9	77.9	82.2	54.7

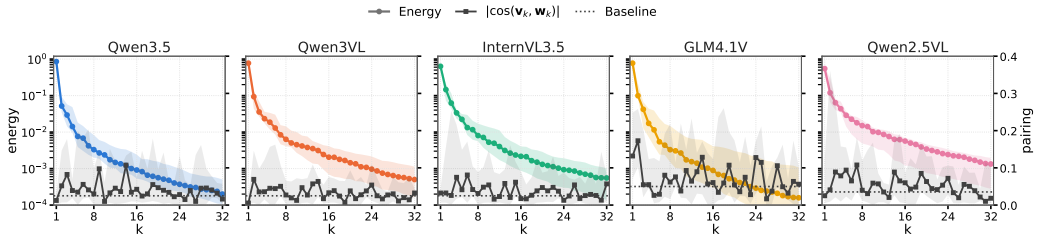


Figure 6: Spectrum of the coupling $\mathbf{C} = \sum_k \sigma_k \mathbf{v}_k \mathbf{w}_k^\top$. Solid line is median over the samples and gray band is the range. We also show the energy, $\sigma_k^2 / \sum_j \sigma_j^2$, of each pair. The dotted line indicates the expected cosine similarity for two random directions in the model's feature dimension.

demanding questions, as shown in Figure 5. We thus assume that Qwen3.5 has the potential to be SOTA with proper regularization on using the prior information, and advocate for using Qwen3.5 as the base model for spatial reasoning to ensure a strong baseline.

B THE PROPERTIES OF U-SPACE

The symmetric coupling contains positive and negative utility directions. Since our goal is to identify features that support the current answer, we define U-Space using the leading positive eigenvalue directions, as in Eq. 4. Negative eigenvalue directions correspond to features whose removal has the opposite first-order effect.

Section 3.2 builds U-Space from the symmetric part of the coupling $\mathbf{C}$ because the singular directions of $\mathbf{C}$ come in near-orthogonal pairs. Here, we show the per-pair coupling strength of the read and write direction $\mathbf{v}$ and $\mathbf{w}$ in Figure 6. We report the median as the solid line as well as the range in the gray area. The content and read directions of every pair are as close to orthogonal as two random directions, i.e. the baseline we show as dotted line.

B.1 RANK OF THE U-SPACE IN RELATIONAL DIRECTION QUESTIONS

In Section 3.3 we choose $K = 64$ for the language-side patching. Here, we show more results by changing K, in Table 6. We can see that models can choose to concentrate the information in as few as rank-16 U-Space such as Qwen3.5, while other models can leave the information more distributed. However, all models have nearly full information ready at $K = 64$, which is still a low-rank subspace in the > 1000 dimensionality full feature space in the VLMs.

Table 6: Flip fraction of the language-side subspace patch against the rank K of the U-Space.

model	layer	K=16	K=32	K=64	complement	random	full state
Qwen3.5	19	0.96	0.92	0.96	0.00	0.00	0.96
Qwen3VL	21	0.24	0.56	0.88	0.00	0.00	0.92
InternVL3.5	21	0.00	0.44	0.78	0.08	0.00	0.86
GLM4.1V	24	0.65	0.83	0.91	0.00	0.00	0.93
Qwen2.5VL	17	0.25	0.67	0.74	0.12	0.00	0.91

Table 7: Left/right patch of the question tokens with the partner question's features restricted to the rank-64 language-side U-Space (language) or to the rank-64 U-Space fitted on the image tokens (vision). We count flip fractions as in Section 3.3, and report the mean principal angle cosine between the two rank-64 spaces.

model	layer	lang./full	vision/full	overlap
Qwen3.5	19	1.00	0.05	0.29
Qwen3VL	21	0.92	0.38	0.32
InternVL3.5	21	0.88	0.36	0.31
GLM4.1V	24	1.00	0.00	0.31
Qwen2.5VL	17	0.89	0.68	0.34

C LANGUAGE-SIDE VISION U-SPACE

The experiment with the results in Figure 2d restricts the partner question's features to the language-side U-Space. A vision-side alternative can take two forms. A transplant on the image tokens themselves is empty by construction: the frames precede the question, so under causal attention their states do not depend on it. This does not make the image tokens irrelevant to the answer. The question tokens read them at every later layer, and an injection that adds a direction to them does move answers; it only means that what distinguishes the two partner questions, which objects stand where relative to the anchor, is not in the image tokens. The other form keeps the patch on the question tokens and replaces the subspace. The restriction comes from rank-64 U-Space of the image tokens rather than the language-side vision residual Δh. Table 7 compares the two subspaces. The vision-side subspace recovers none in GLM4.1V and .05 in Qwen3.5, where its complement carries almost the whole effect, .36 and .38 in InternVL3.5 and Qwen3VL, and .68 in Qwen2.5VL. The two spaces overlap only partly, with a mean principal angle cosine .29 to .34. A language-side operator is thus necessary for composing the vision-side patch-level information in relational questions.

D FRONT/BACK: THE SCENE-LEVEL AXIS

The front/back axis of Section 3.3 is in all five models. However, we find that the front/back answers suffer from a dataset bias. 77% of front/back answers in VSI590K and 84% in VSI-Bench are "front", whereas left/right is balanced. The mechanism is therefore different from the left/right one, in that the front/back answer relies on a fixed vision-side axis whose positive end is the "front" answer. We take 40 questions from the relative direction questions in VSI-bench, 20 whose answer involves "front" and 20 whose answer contains "back". We use U-probe on the vision side to get U-Space, on 39 of the 40 questions. We inject the top-1 direction of that U-Space into the remaining one question's image token features, using the operation in the steering experiment. We find that all injections with a positive steering weight changes the answer towards "front", while when we repeat the experiment on the left/right case a positive steering weight changes the answer towards what the model has already generated. Intuitively, the axis is fitted on questions balanced between the two answers, so it can only be shared if the model reads the scene alike across questions. For front/back it does: the same feature raises the front score in front-answer and back-answer questions alike, and the shared direction is a push that moves every answer the same way. For left/right it does not: the same wall or object raises the left score under one observer and target and lowers it under another,

Table 8: Front/back patching for steering. We show the number of questions n, and the one that flips under the full-feature patching "full". We report the flips under U-64, cpl, and r-64 just like in Section 3.3, along with the ratios. The last two columns split the count by direction as n/full/U-64, where n is the number of the question with the answer on the left of the arrow. They show that all back answers are flipped to front under the patching, while the "front" answer, suffered from a prior, resists to become "back" even under the full-feature patching.

model	U-64/full	cpl/full	front→back	back→front
Qwen3.5	0.98	0.00	42/26/25	28/28/28
Qwen3VL	0.93	0.07	39/19/17	26/26/25
InternVL3.5	0.92	0.20	40/11/9	14/14/14
GLM4.1V	0.94	0.00	41/14/12	19/19/19
Qwen2.5VL	0.78	0.26	42/12/7	11/11/11

so the question-specific reads cancel, and what remains can at most sharpen or weaken the decision each question has already made.

Facing reversal. On the same questions, rewriting "facing the Y" as "facing away from the Y" turns the observer by 180 degrees, which negates both components of the correct answer. Note that however, the models are not supposed to be adapted to such an out-of-distribution condition and will collapse towards a prior or random baselines (Zhang et al., 2026c). The generated answers collapse to the frontal default in every model, whereas the left/right answers stay divided between the two sides. The front/back component does not follow the stated frame, and the left/right component does not collapse towards a fixed prior.

D.1 Front/back from the language side

The left/right patch of Section 3.3 has a front/back counterpart. We repeat the left/right patching experiment in Section 3.3 and find that patching U-64 only carries most of the full-feature flips, within a ratio of $[0.78, 0.98$ across 5 VLMs. Detailed results are in Table 8. An interesting fact is that a back answer moves to front in all cases under full patching, while a front one moves to a back in only 28% to 62% of the cases under full patching. It indicates that the vision-side fixed front prior in Appendix D plays a role during the patching, and the model relies on that dataset prior to answer.

E More results of interpretability

We show more visualizations and logit lens results of the interpretability part in Figure 7, 8, 9, 10, 11. An eigenvector is defined only up to its sign, so which end of an axis projects high is arbitrary. We therefore identify the two ends of an axis by what they show. For every axis we take the patches with the highest and with the lowest projection, and we orient the axis by content. The orientation decides on which side a set of patches is drawn, not which patches are selected, and it is the orientation under which the same content recurs across models. We note that many patches lie on the wall or floor area, motivating our experiments on testing that these scene structure areas are important for spatial reasoning.

F Details of cross-dataset consensus of U-Space

We measure the cross-dataset similarity using principal angle cosines between rank-K U-Spaces on VSI590K and SPAR. We choose $K = 4$ because we find that within-dataset split-half ceiling of the similarity is around 0.6 on the two datasets, a moderate value. The answer-used features are not the same across questions as different samples require different knowledge in the scene, while they share certain similarity as spatial reasoning needs common scene features. We choose 50 questions for each question type. Changing the samples does not make huge difference in the result. Table 9 gives the results. The U-Spaces are similar between the two datasets on partner question types. Different types of questions require different visual knowledge, but a large part of it is shared: the similarity to the non-partner types of the other dataset stays high while being lower than the partner one, in line with Section 3.4.

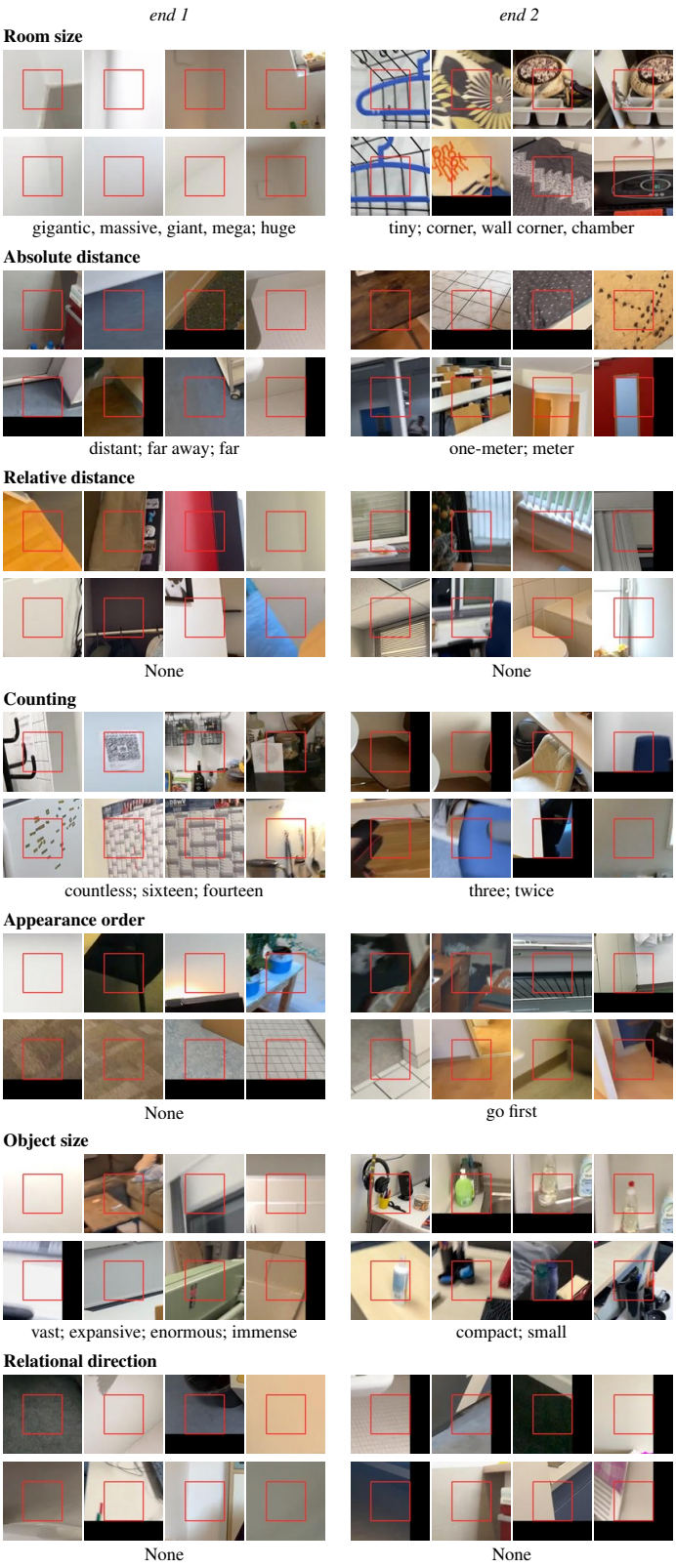


Figure 7: Patches of the leading U-Space axis per VSI-Bench type, Qwen3.5.

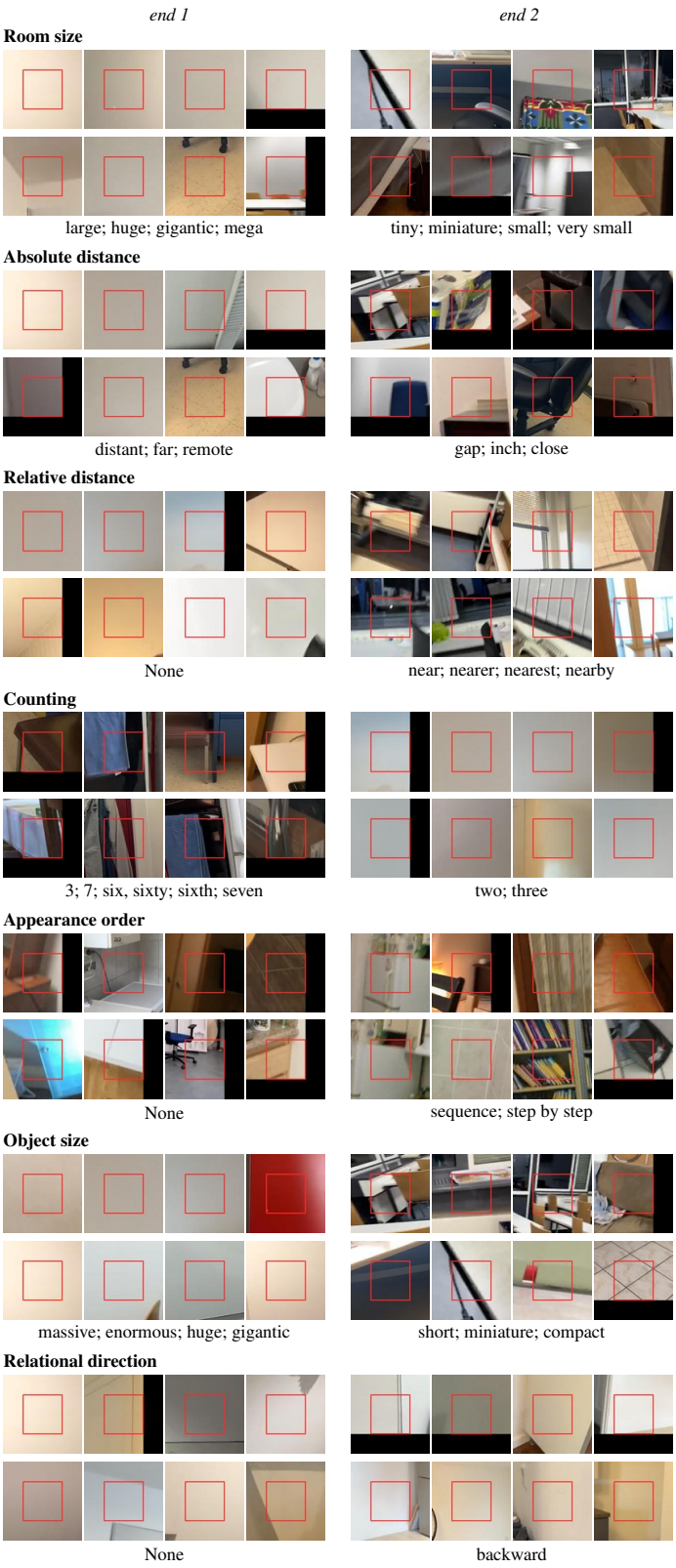


Figure 8: Patches of the leading U-Space axis per VSI-Bench type, Qwen3VL.

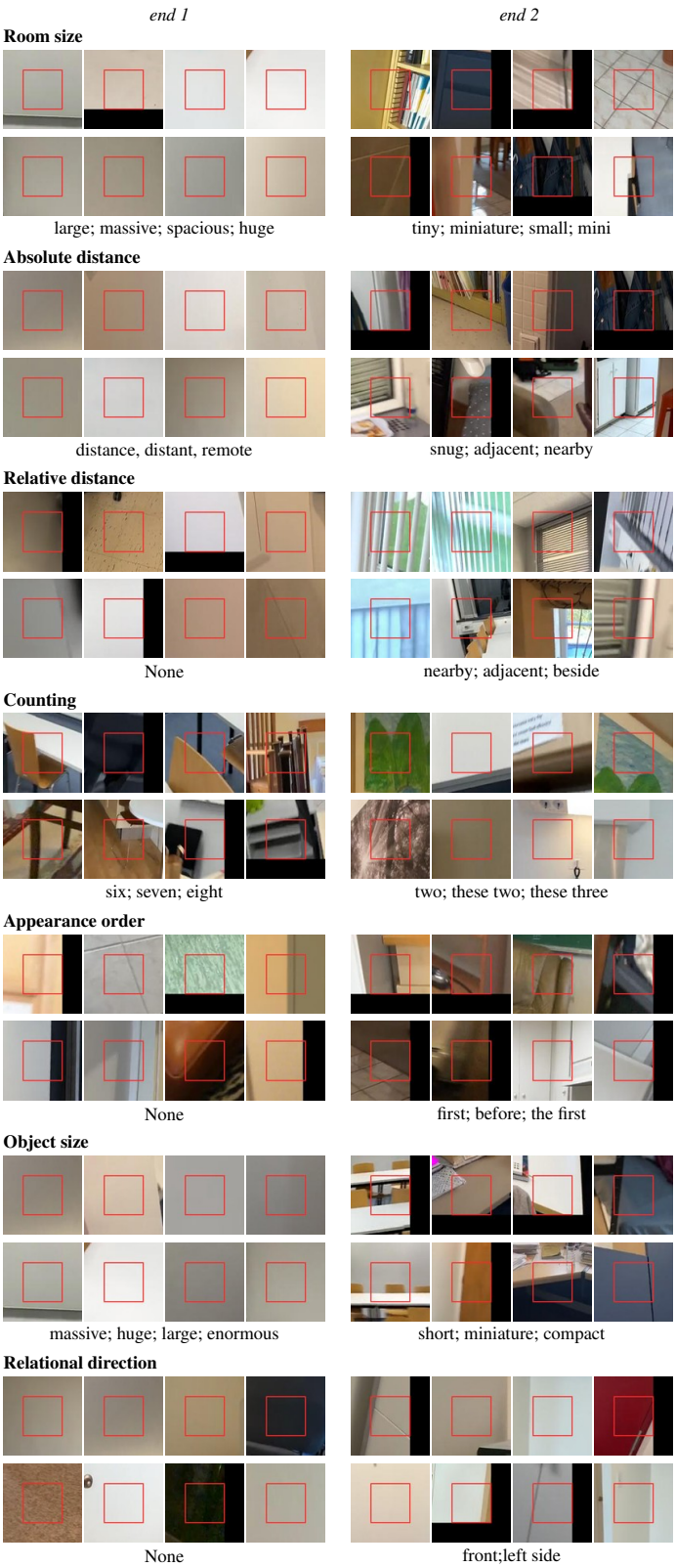


Figure 9: Patches of the leading U-Space axis per VSI-Bench type, InternVL3.5.

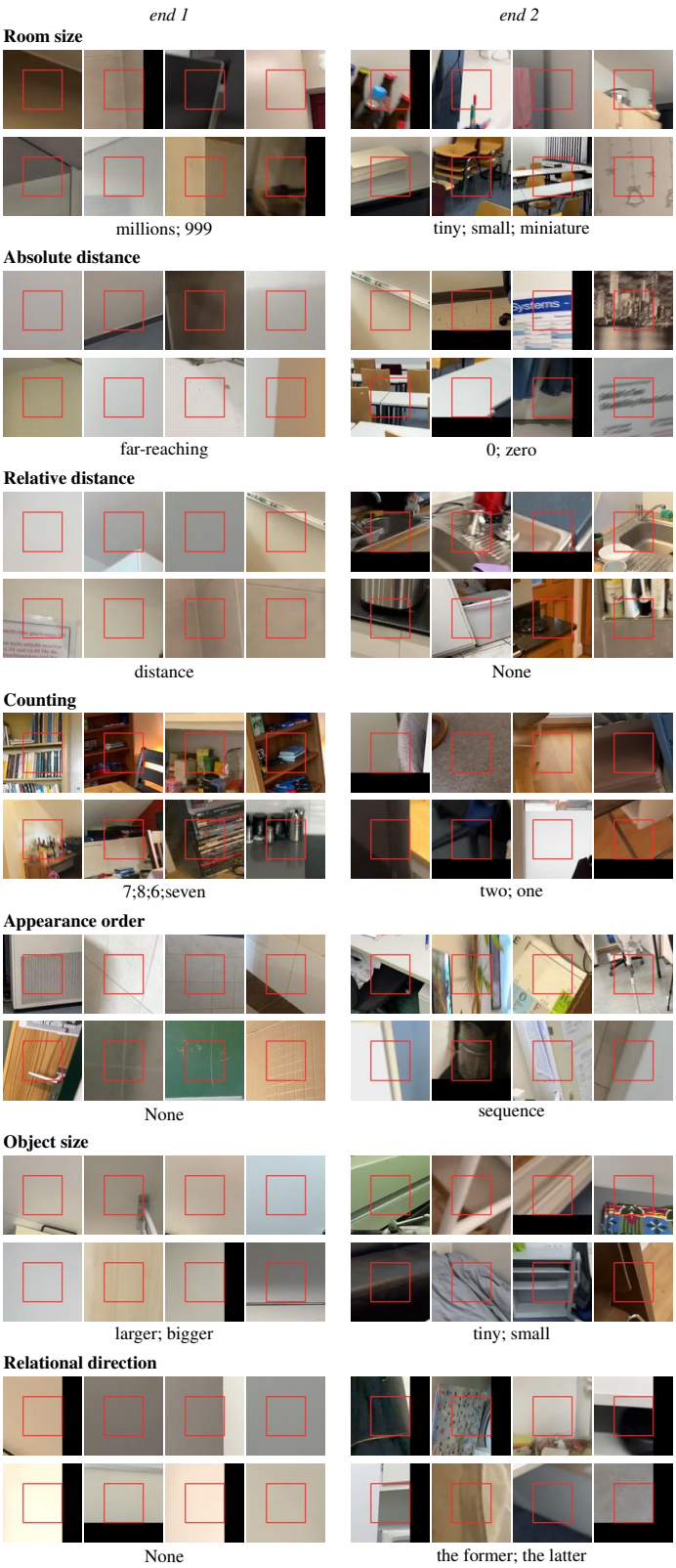


Figure 10: Patches of the leading U-Space axis per VSI-Bench type, GLM4.1V.

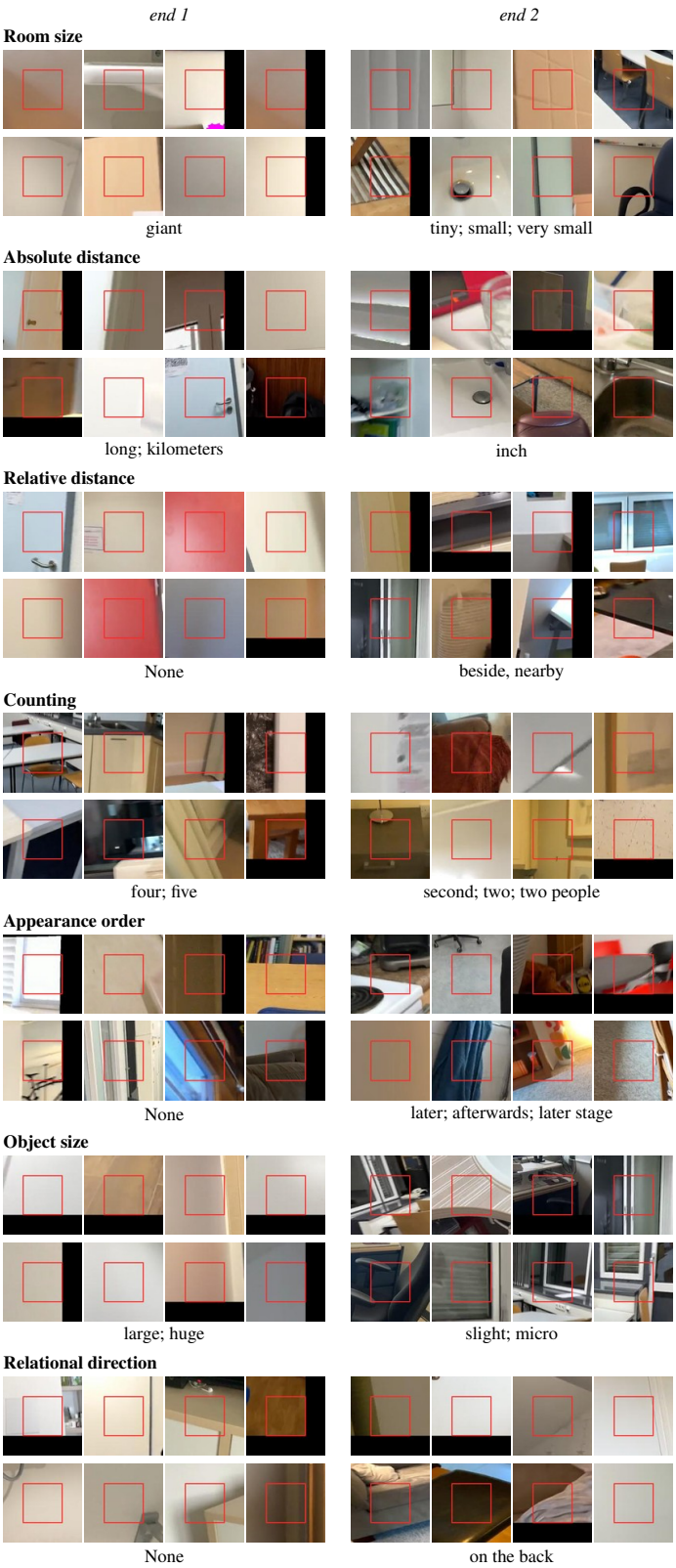


Figure 11: Patches of the leading U-Space axis per VSI-Bench type, Qwen2.5VL.

Table 9: Cross-dataset similarity of rank-4 U-Spaces, measured by principal angles cosine, divided by the split-half ceiling within each dataset. We report mean ± standard deviation over the five VLMs. The partner is the type with the highest similarity with the VSI type, and non-partner means averaging the similarity over all other question types than the partner. abs-dist, rel-dist: absolute and relative distance; rel-dir: relational direction; dist-pred: distance prediction; spatial imag: spatial imagination; appearance: appearance order.

VSI-Bench type	SPAR partner	partner	non-partner
abs-dist	dist-pred	0.78 ± 0.11	0.58 ± 0.05
rel-dist	dist-infer	0.79 ± 0.09	0.59 ± 0.05
room	room	0.68 ± 0.04	0.45 ± 0.02
rel-dir	spatial imag	0.72 ± 0.03	0.61 ± 0.04
counting	counting	0.83 ± 0.11	0.54 ± 0.02
appearance	appearance	0.70 ± 0.08	0.53 ± 0.06

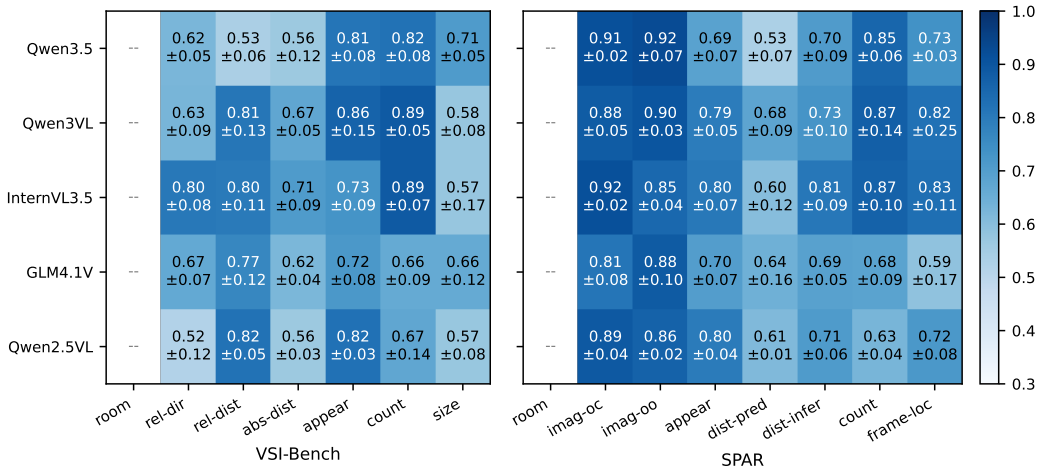


Figure 12: The object in the question is intuitively necessary. Each cell is the fraction of questions of a type on which deleting the type's rank-4 U-Space on the `queried objects` tokens lowers the answer score more than the same deletion on a size-matched random token set.

The U-Space is found even where the model fails. VSI590K SFT is known not to transfer to MMSI (Lin et al., 2025; Yang et al., 2026a) and the model behaves at chance-level, yet we still find the U-Space on similar question types to be coherent. The difference is that the eigenvalues of $\mathbf{S}$ is 20-times smaller than the one on VSI or SPAR, indicating a lack of the answer-content coupling.

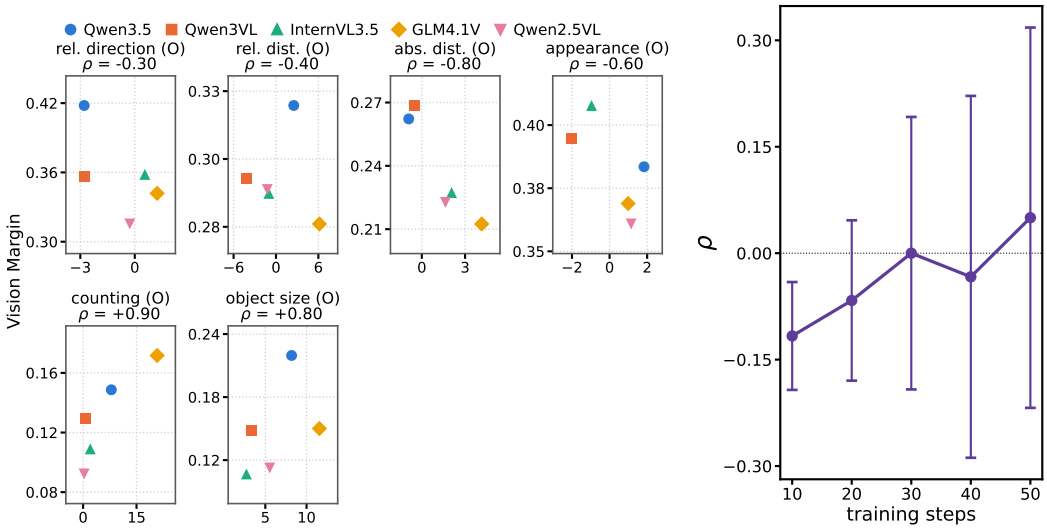
G Object matters in all object-involved questions

We perform the regional deletion in Section 4.1 and claim that `queried objects` region matters in all question types that involve objects. We show the results in Figure 12, and they support our claim.

H More results of the predictor

H.1 Building the predictor based on object or others region

We show the results of the per-category utility change on `queried objects` and `others` region, and the corresponding vision margin. The results are shown in Figure 13, 14. The aggregated correlation is never statistically significant, and utility change in these two regions shows reverse trend against the vision margin on the scene-demanding types, e.g., "room size", "relational direction", "relational distance". We can see an interesting phenomenon that models tend to allocate



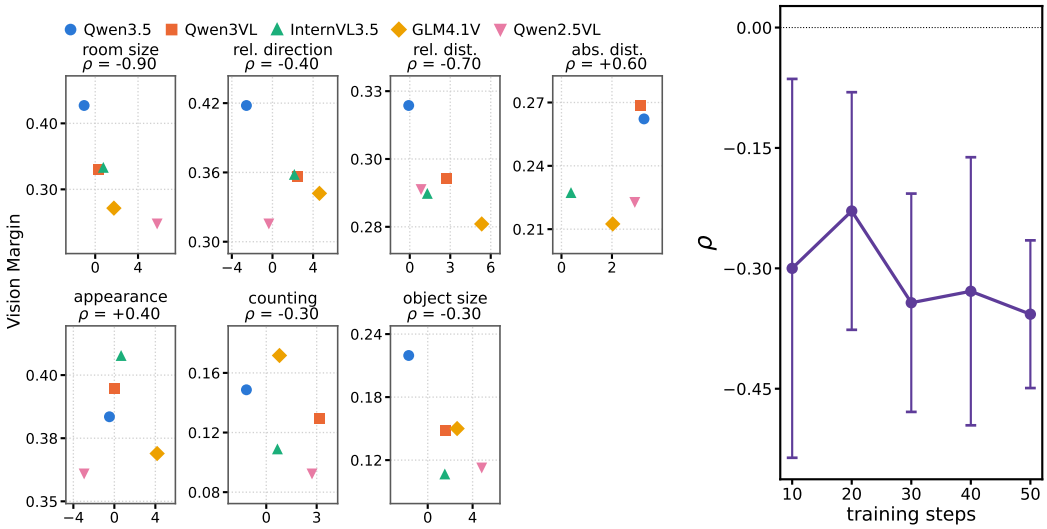
(a) (b)

Figure 13: The predictor on VSI, with `queried objects` as the reading region. Note that "room size" is not applicable in object-based predictor. (a) X is the change of the region's fraction of $\sum_p |u_p|$ from the base model to step 20 of SFT, using the model's own generated answer. Y is the vision margin at step 800 on that type. ρ is the five-model Spearman correlation. (b) The aggregated ρ over all question types at each SFT step. The aggregation is tested under the null hypothesis that model labels are exchangeable.

utility on `others` more, in that its regional utility fraction grows positively. We assume that this is due to the object-centric nature of VLM pretraining, in that a large portion of the data is about object properties and grounding. The `others` region includes exactly natural objects, yet none of them is relevant to the question. However, the model tends to recognize those objects because it costs less energy to focus on them than to focus on plain walls and floors. This interpretation remains a hypothesis and we leave the validation future work. The fact that the `others` region growth and the model performance shows a reverse trend indicates that growth of utility fraction on the `others` region is negatively associated with final vision margin on scene-demanding types. The model might need to refrain from the incentive to attend to object in spatial reasoning but allocate the utility to `scene structure`.

H.2 Obtaining statistical significance

For a SFT step m and a question type t, the predictor gives one number per model, $X_{m,t}$, the change of the region's fraction of $\sum_p |u_p|$ from the base model to step m, and the outcome gives another, $Y_{m,t}$, the vision margin at step 800. With five models, the per-type Spearman correlation,$\rho_t = \mathrm{Spearman}(X_{\cdot,t}, Y_{\cdot,t})$, is highly discrete and unstable so we do not interpret individual ρ_t. The aggregated statistic is the mean over the T types, $\bar{\rho} = \frac{1}{T}\sum_t \rho_t$, with $T = 7$ on VSI-Bench and SPAR. The null hypothesis is that a model's profile of X across the types carries no information about its profile of Y, that is, the model labels are exchangeable. Under this null, every one of the $5! = 120$ assignments of the five models to the five Y profiles is equally likely, so the reference distribution can be enumerated exactly. For each permutation π of the five labels we re-order the Y vectors by π, the same π for every type, recompute every ρ_t and their mean $\bar{\rho}_\pi$, and report $p = \frac{1}{120}\#\{\pi : \bar{\rho}_\pi \geq \bar{\rho}\}$, so that $p \geq 1/120 = .0083$. This is how we get the floor of $p = 0.008$, and p-values have an interval of $1/120$. Applying one permutation to every type is what makes the T types a single test rather than T tests. The T can be interpreted as repeated measurements of the same five experimental units, rather than additional experimental units. The types share the same five models so they are not independent replicates.



(a) (b)

Figure 14: The predictor on VSI, with `others` as the reading region. (a) X is the change of the region's fraction of $\sum_p |u_p|$ from the base model to step 20 of SFT, using the model's own generated answer. Y is the vision margin at step 800 on that type. ρ is the five-model Spearman correlation. (b) The aggregated ρ over all question types at each SFT step. The aggregation is tested under the null hypothesis that model labels are exchangeable.

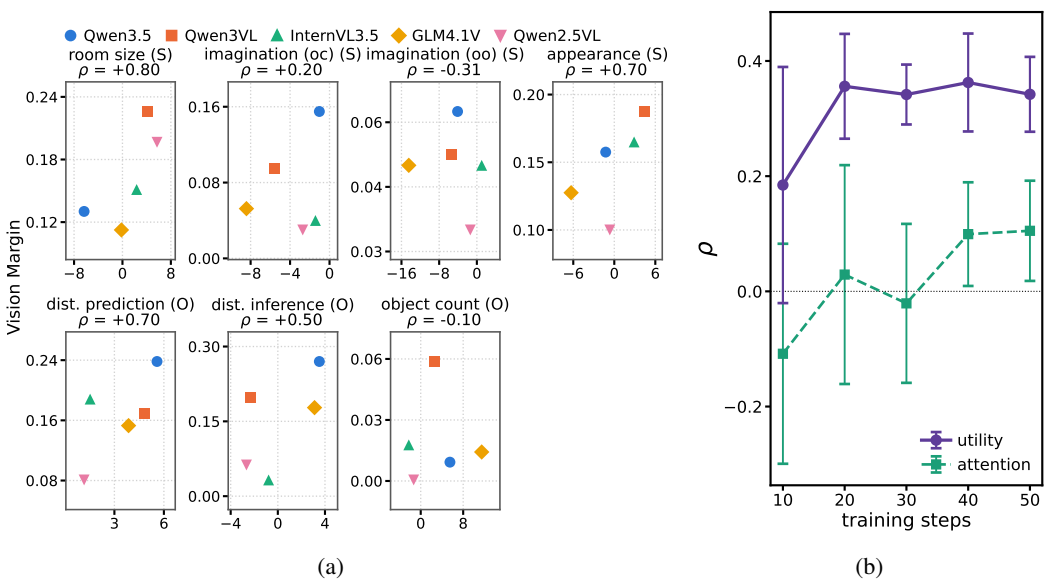
H.3 Predictor on SPAR

We present the results of the predictor on SPAR, in Figure 15. We note that the training on SPAR does not work as well as on VSI590K, as the vision margin is low in some categories. We drop the "frame-loc" as some questions with this type has a "none" answer, that is no object can be located in the ground truth. We also notice that 54% answer of the "counting" type is 2, so this is a prior-contaminated category. Due to the poor training, the predictor performance does not match the one on VSI, yet the trend still holds. The utility predicts after 20 SFT steps with p-value in the five points: $0.13, 0.05, 0.06, 0.02, 0.02$. The attention-based predictor behaves at chance-level, and no p-value is meaningful. We use this predictor result as a support of our claim that spatial reasoning runs on scene structure in scene-demanding questions, instead of delivering another well-performed models after spatial SFT.

Table 10: SPAR validation accuracy at step 800 (%). imag: spatial imagination (oc: object-centric, oo: object-to-object); dist-pred: distance prediction; dist-infer: distance inference from a center point.

model	overall	room	imag-oc	imag-oo	appear	dist-pred	dist-infer	count
Qwen3.5	54.2	46.1	76.5	68.2	19.0	54.7	65.0	59.2
Qwen3VL	53.6	55.8	71.2	66.0	24.2	51.0	57.8	59.6
InternVL3.5	43.7	45.7	63.7	63.2	21.5	43.7	42.5	56.2
GLM4.1V	49.2	44.0	67.5	67.2	19.0	50.9	56.5	57.2
Qwen2.5VL	43.7	50.5	65.2	63.7	15.0	41.4	45.8	56.5

Note that we assign two distance inference question types with O, `queried objects`, as the region to be read. This is because we find that in these two categories, although the model indeed needs `scene structure` as shown in Figure 4 in the main paper, the predictor needs to read the object re-appearance of the marked out object in other frames. We show an example in Figure. Our reading is that a point marker covers a few tokens and identifies nothing by itself, so before the model can measure a distance or a depth it has to find the same object in the other frames; the binding of an instance across views replaces the anchoring of a named object to the room that the VSI



(a) (b)

Figure 15: The predictor on SPAR. (a) X is the change of the region's fraction of $\sum_p |u_p|$ from the base model to step 20 of SFT, using the model's own generated answer. The region is scene structure (S) on the scene types and the queried objects (O) on counting, and two question types with explicit object markers (see the text). Y is the vision margin at step 800 on that type. ρ is the five-model Spearman correlation, on which we do not make any interpretation. (b) The aggregated ρ over all question types at each SFT step. The aggregation is tested under the null hypothesis that model labels are exchangeable.

questions require. To validate, we run our predictor on the multi-view SPAR dataset (Zhang et al., 2026a), that includes five point-marker types and three bounding-box types. The point-marker types are depth prediction, object-centric (depth-oc) and object-to-object (depth-oo), distance inference from a center point (dist-infer), and distance prediction, object-centric (dist-pred-oc) and object-to-object (dist-pred-oo); the bounding-box types are object spatial relation, object-to-object (sp-rel-oo), and spatial imagination, object-centric (imag-oc) and object-to-object (imag-oo). All the question types contain "mv" in the type name, indicating the multi-view property. They are all asking about scene information, yet we assume that the point marker question needs the object re-appearance reading as the performance predictor. We train the 5 VLMs on a subset of SPAR that contains the multi-view questions, using the same setting in Appendix A. We benchmark the trained models on the corresponding question types in SPAR-Bench (Zhang et al., 2026a). To locate the referent in the frames, we use SAM3 (Carion et al., 2026) prompted with the object names, restricted to the non-marked images. On our video set, where the object re-appears across 32 frames, we track the object from its marked frame with SAM2 (Ravi et al., 2024). The per-category result is shown in Figure 18. Under the rule that point marker types use object re-appearance regions and the other types use `scene structure` as usual, the per-category utility change shows a positive trend with the final performance. Note that, however, the marked object alone does not predict. It is the re-appearance that does the tracking of the performance. This result not only makes our spatial reasoning interpretability more complete, but also proves that our results and conclusion generalize to multi-view datasets, in that Qwen3.5 achieves the best results on both SPAR datasets, the video one and the multi-view one.

H.4 Training loss disagrees with regional utility change

We test the training cross-entropy loss and find that Qwen2.5VL achieves good losses during SFT and at step 20 the training loss matches that of Qwen3VL. Our predictor at step 20, in contrast, ranks the Qwen2.5VL at last. We also test the 7-type aggregated correlation based on training loss and it is not statistically significant, giving $\rho = 0.39$ with $p = 0.125$ at step 20. We further test the training set accuracy on the data subsets that we use to build the predictor, and it gives $\rho = 0.42$ with



Figure 16: The re-appearance of the objects that are marked out using point markers, in the SPAR video dataset we derived from SPAR-7M.



Figure 17: The re-appearance of the objects that are marked out using point markers, in the SPAR multi-view dataset we derived from SPAR-7M.

$p = 0.092$. The problem with training set accuracy is that models can collapse to similar training accuracy during SFT but have different final performance. For example, at step 20, the accuracy of all models except Qwen3.5 on relational direction questions is around 38 to 40. On "counting", the standard deviation of accuracy across 5 models is just around 0.03. The correlation signal can also differ significantly across subsets. For example, at step 20, the three subsets give 0.61, 0.49 and 0.22. Note that the loss or accuracy change from the base model is not applicable in some models, as the loss can be very large (e.g., 14 in GLM4.1V at step 0 compared to around 0.5 at step 800) on the thinking models when adapting to the direct output QA format from generating the thinking tokens. For the same reason, the base model accuracy is also suffered from the format or out-of-distribution issue. We highlight that, although obtainable, the training loss and accuracy both need the ground truth labels while our predictor uses label-free, model's generated answer, broadening its application range to scenarios where ground truth is not available at all while one needs to estimate the model performance in spatial understanding. Training set metrics do not necessarily reflect the test performance. They order models more weakly and less stably than our regional-utility-based predictor, which gives statistically significant prediction of the final model performance on held-out data.

H.5 Application of the diagnostic predictor to a new model

The predictor of Section 4.2 is fitted on five models. We use our conclusion and the predictor to diagnose the SFT of a new model, Gemma4-12B (google/gemma-4-12B-it) (Team et al., 2026). We use the same training recipe as in Appendix A, and record the utility fraction on `scene structure`. We do not perform feature resize, but use the soft token limit specific to Gemma to control the token number. We find that Gemma4 does not use the scene structure during SFT, as shown in Table 11. According to our predictor, Gemma4 is predicted to be the worst model in this case, despite being the largest model. It indeed does not perform well on VSI-Bench, achieving 39.7 overall accuracy. The base model accuracy is 19.7, with refusals to answering questions, e.g., "I cannot provide an estimate of the room's size from these images". The SFT is thus working well, but does not yield a good model at the end. This could either be due to the soft token limitation being too low (we use 70 to match the token budget in other models), the SFT configuration being not suitable, or the Gemma model itself does not have good features for spatial understanding. Whatever the cause, the label-free predictor at places Gemma-4 where its 800-step accuracy ends, acting as an early diagnostic of the Gemma4 failure on VSI-Bench.

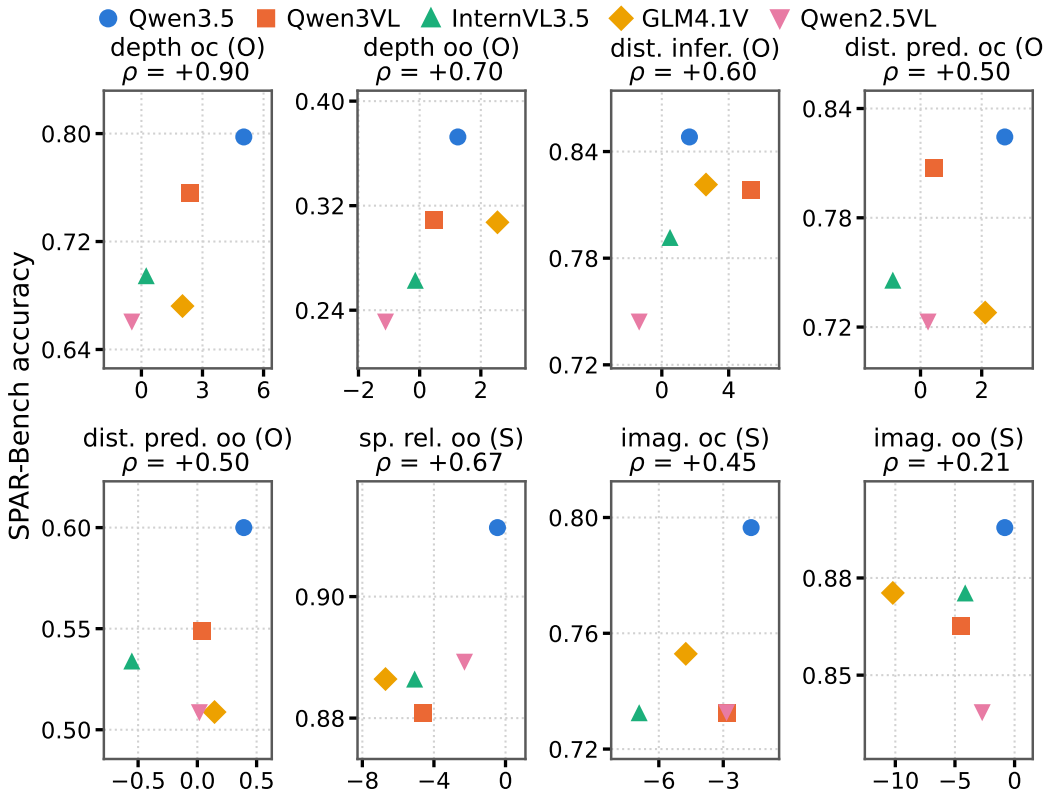


Figure 18: The predictor on SPAR multi-view dataset. Note that this is not the SPAR video set we use in the main paper, but a 2-4 image subset of the SPAR-7m data. It further generalizes the applicability of our predictor onto multi-view dataset beyond video-based data. X is the change of the region's fraction of $\sum_p |u_p|$ from the base model to step 20 of SFT, using the model's own generated answer. The region is the queried objects (O) on all types with a point marker that marks out the queried objects, and scene structure (S) on the other scene types. Y is the SPAR-bench accuracy at step 800 on that type. ρ is the five-model Spearman correlation.

Table 11: Change of the `scene structure` fraction from the base model (mean over the five scene types) and of the object on counting and object size, on the model's own answer, and the vision margin at step 800. Gemma4 has the largest drop of `scene structure` fraction, and it is indeed a poor model on VSI-Bench. The predictor successfully acts as an early diagnostic of the Gemma4 failure.

Model	step	Δ structure share (scene)	margin Y (scene)
Qwen3.5	50	+0.010	+0.36
Qwen3VL	50	+0.006	+0.33
InternVL3.5	50	−0.015	+0.32
Qwen2.5VL	50	−0.023	+0.28
GLM4.1V	50	−0.072	+0.26
Gemma-4-12B	10	−0.031	+0.05
Gemma-4-12B	20	−0.067	+0.05
Gemma-4-12B	30	−0.044	+0.05
Gemma-4-12B	40	−0.088	+0.05
Gemma-4-12B	50	−0.144	+0.05

I TEST-TIME INTERVENTIONS BASED ON SCENE STRUCTURE

Section 3.3 showed that U-Space perturbations move the generated answer. This section records that the two test-time uses of that causal handle we tried do not raise benchmark accuracy.

Steering. We try injecting U-Space axes at test time, yet the benchmark accuracy barely changes. The axes in U-Space move the answers in the same direction, while a correction of the answer needs a signed move per-question, which is what the model's reading computes. That reading is exactly learned during SFT, while the U-Space supplies the feature direction.

Re-weighting attention onto `scene structure`. We try multiplying a factor c onto the attention weights of the textttscene structure area, implemented as $+\log c$ on their pre-softmax logits, yet the benchmark accuracy barely changes or can be even worse. This is due to the complex encoding nature of the token features, that a image token is a mixture over appearance, spatial, geometry features. A single routing scalar cannot select what a token carries, and changing the scalar amplifies or attenuates every component at once.

J THE STRUCTURE SHARE AS A TRAINING OBJECTIVE

The objective the predictor suggests. Section 4.2 invites a regularizer: the models that keep more of their utility on `scene structure` during finetuning end with larger vision margins, so a term that stops the share from falling should help. We test this on InternVL3.5. For every training question of a scene type (room size, absolute and relative distance, relational direction, appearance order) the loss adds $\lambda \sum_{l\in\{17,26\}} \max(0,\, \bar{s}_{t,l} - s_l)$ to the cross-entropy, where $s_l = \sum_{p\in\Omega} |u_p| / \sum_p |u_p|$ is the share the predictor reads, with $u_p = \mathbf{h}_p^\top \mathbf{g}_p$ of layer l and Ω on the `scene structure` tokens, and $\bar{s}_{t,l}$ is the base model's mean share on type t; counting and object size carry no term. Only the projector is trained (34M parameters; vision encoder and language model frozen), for 100 steps at batch 256 from the base model, with $\lambda = 2$ and $\lambda = 8$. We only train the projector as adding LLM training would not fit the GPU memory as it involves second-order optimization. The term is active on about 60% of the scene questions, yet it does not move its target: on the same questions in the same order, increasing λ changes the share by $+.0001$ at layer 17 and by $+.0002$ at layer 26. VSI-Bench accuracy is .5642 without the term and .5668 and .5675 with it; on the scene types the per-question difference is $+.005$ (95% interval $-.002$ to $+.012$) and $+.007$ ($-.001$ to $+.014$), on counting and object size $-.001$. The run without the term stays at the base model's accuracy (.566), so 100 projector steps are a small perturbation of the base model, and we read the run as a test of the term, not of its ceiling. We do not scale it up, for reasons that lie in what the fraction is rather than in these numbers.

Why the share is a diagnostic and not an objective. Our predictor orders base models; it is not an improvement metric within one. The five models are finetuned on the same data, with the same frames and number of steps, and they are already ordered at step 20 by how much of the utility fraction they keep (Section 4.2); the order is therefore largely a property of the models that enter SFT. What the predictor rewards is losing less of the scene structure utility, not gaining it, and nothing in it says that pushing one model's scene structure utility up makes that model better. SFT, moreover, redirects the read and inherits the content (Section 3.6). A read can only pick up what the tokens carry, and whether the scene structure tokens carry the extent and the layout of the room in a readable form is, on this evidence, a property of the pretrained model. In our run the term could not move the regional utility through the projector, the one component through which SFT can rewrite that content without retraining the language model. We therefore see the utility as a criterion for the stage at which that content is formed: for choosing a model at the early stage of SFT as a quantity to monitor, or as a guidance to optimize during pretraining, where the features of the scene are learned rather than read.