FROM SYMBOLIC MAPPING FAILURE TO NUMEROSITY SPECIALIZATION IN VISION-LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Visual number sense supports counting objects, comparing group sizes, and determining the size of selected or combined groups. Vision-language models exhibit sensitivity to quantity, but their ability to answer numerical questions remains uneven across visual scenes and tasks. We introduce a two-phase post-training framework to develop more accurate and reusable visual number sense in pretrained models. Configuration Coverage first associates shared count labels with complementary object layouts and appearances. A joint update then combines Relational Numerosity Supervision, which links counts through local changes and part–whole relations, with multi-interface supervision for comparison and query-conditioned counting. We also construct a controlled synthetic dataset for training and evaluation across visual configurations and count ranges. Experiments on two VLM families show substantial improvements in synthetic counting and task-dependent gains on external numerical benchmarks. Matched-budget ablations establish the benefit of complementary configuration coverage, while supervision-allocation studies demonstrate the contributions of relational and task-specific training. Representation analyses reveal more accurate intermediate-layer count readout and stronger cross-configuration readout transfer. We characterize the resulting profile as *numerosity specialization*: high-precision counting develops within targeted visual and numerical regimes, with varying transfer to larger cardinalities, other visual distributions, and numerical tasks.

1 INTRODUCTION

Numerosity provides a basic link between visual perception and symbolic reasoning. Recognizing how many objects are present supports comparison, composition, and decisions about query-defined sets. For vision-language models (VLMs), reliable numerical capability therefore requires both representing quantity and expressing it through the appropriate numerical response. These abilities can diverge: by distinguishing visual individuation, magnitude awareness, and symbolic mapping, Pang et al. (2026) show that quantity information can remain readable from model representations even when numerical generation deteriorates. This gap motivates a constructive question: how should post-training supervision be organized to develop more accurate and reusable numerical capability?

The challenge extends beyond learning an image-to-number mapping on a single distribution. The same cardinality can appear in substantially different arrangements and visual forms, while different cardinalities must become connected to support larger-count and compositional operations. Quantity must also remain usable when a query changes from direct counting to comparison or conditional selection. We study these requirements through *functional numerosity access*: the ability to produce the correct numerical response for a scene and query. We organize this capability along visual configuration c, task-relevant cardinality n, and task interface τ (Figure 1). These coordinates identify what supervision needs to support: stable numerical responses across visual realizations, extension across numerical range, and reuse under different task demands.

We focus on strengthening numerical capabilities in existing pretrained VLMs through targeted post-training, building on their learned visual and language representations. Human numerical cognition offers a useful design intuition. Subitizing supports precise enumeration of small sets, while grouping and structured enumeration facilitate the processing of larger quantities (Trick & Pylyshyn, 1994; Wege et al., 2022). Motivated by these observations, we organize supervision from

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.

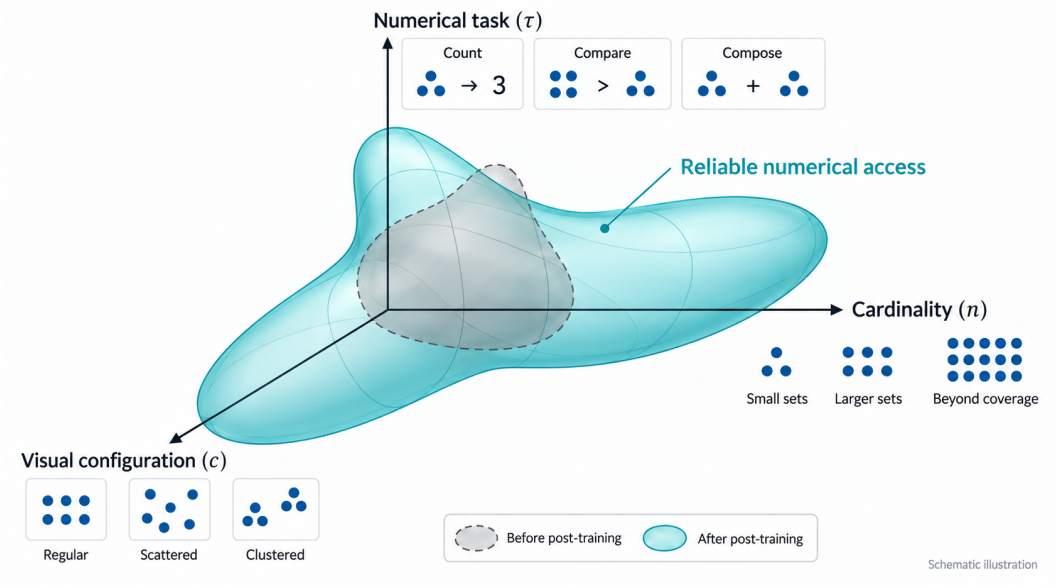


Figure 1: **Three dimensions of functional numerosity access.** Visual configuration c describes the arrangement and appearance of a set; cardinality n specifies its task-relevant size; and task interface τ specifies how quantity is queried or used. Gray and teal schematic regions illustrate how post-training can expand and reshape reliable numerical behavior. Configuration and task coordinates index categorical families.

configuration-robust counting toward larger-count, comparison, and set-based tasks. *Configuration Coverage* groups complementary visual realizations under shared cardinality labels, combining systematic occupancy patterns with varied layouts and appearances. Its organizing principle is to expose the same numerical target across different forms of visual variation. *Relational Numerosity Supervision* (RNS) connects cardinalities through direct-count examples, local numerical changes, and part–whole relations. *Multi-interface supervision* trains models to compare group sizes, count objects with specified attributes, and count objects in unions or intersections of sets. We implement these components in two optimization phases: Coverage initialization followed by a joint RNS and multi-interface update.

We construct a controlled synthetic dataset spanning visual configurations and count ranges. On Qwen3-VL-8B-Instruct, the full framework reaches 99.66% accuracy at counts 5–8 and 85.42% at counts 9–12. Under the same downstream sample budget, reallocating interface supervision to RNS raises 9–12 accuracy from 27.78% to 74.65%, while Coverage initialization further improves it to 85.42%. Similar gains appear on the synthetic OOD evaluation and InternVL3-9B-Instruct, with smaller improvements on external numerical benchmarks. Matched-budget studies further show that combining enumerative and factorized Coverage improves robustness, while relational and interface supervision support higher-count acquisition and task-specific numerical use.

Complementary analyses examine the internal changes accompanying this training path. On Qwen, Coverage makes quantity more accurately decodable from intermediate representations and improves the transfer of count-specific prototypes from Grid to Scattered and Clustered layouts without refitting. Joint relational and multi-interface training retains much of this readout advantage while recovering generated-count accuracy relative to the Coverage checkpoint on the same prototype-target images. Evaluating these measures together shows that internal numerical readout and generated numerical behavior can evolve differently during post-training.

We characterize the overall profile as *numerosity specialization*: high-precision counting develops within targeted visual and numerical regimes, with varying transfer to larger cardinalities, other visual distributions, and numerical tasks. This perspective connects the organization of supervision to both the development of numerical capability and the conditions under which it can be reused.

Our contributions are:

- **A structured formulation and post-training framework.** We organize functional numerosity access along configuration, cardinality, and task interface, and develop a two-phase framework combining Coverage, relational, and multi-interface supervision.
- **A controlled visual numerosity dataset and evaluation suite.** We construct synthetic training and evaluation sets spanning visual configurations, count ranges, and numerical queries, enabling controlled evaluation of counting accuracy, robustness, and transfer.
- **Behavioral and representational evidence.** Across two VLM families, our framework improves counting and transfer to held-out visual distributions while largely preserving task-specific numerical performance. On Qwen, representation analyses further reveal stronger internal quantity readout.

2 Related Work

Numerical cognition and subitizing. Human numerical cognition combines precise judgments of small sets with approximate estimation of larger quantities. Classical accounts distinguish object-based and approximate magnitude representations (Feigenson et al., 2004), while other work explains both regimes through a shared capacity-limited representation (Cheyette & Piantadosi, 2020). Visual organization also shapes enumeration: familiar configurations can facilitate subitizing (Mandler & Shebo, 1982), and grouping cues improve numerosity estimation through *groupitizing* (Anobile et al., 2020). Cross-notation adaptation further suggests shared magnitude coding between dot arrays and numerical symbols (Piazza et al., 2007). We do not assume that VLMs implement the same mechanisms; rather, these findings motivate studying numerical capability through visual configuration, cardinality, and symbolic access.

Counting in vision-language models. Prior work has examined both the failure modes and internal mechanisms of VLM counting. Campbell et al. (2024) relate multi-object failures to the binding problem, while Pang et al. (2026) distinguish visual individuation, magnitude awareness, and symbolic mapping, showing that quantity information can remain internally available even when numerical generation fails. CountScope combines behavioral, probing, and causal analyses of layerwise counting processes (Hasani et al., 2026), and NumerosityVLM uses controlled visual factors and layerwise probes to study numerosity representations (Fu et al., 2026). These studies motivate separating internally readable quantity from reliable numerical output.

Complementary work improves counting through targeted objectives, data, or model interventions. Paiss et al. (2023) introduce a counting-contrastive objective for CLIP; SITUATE uses controlled synthetic scenes for VLM fine-tuning and natural-image transfer (Peinl et al., 2026); Reason-RFT applies reinforcement fine-tuning to visual reasoning tasks including counting (Tan et al., 2026); and Counting Circuits combines mechanistic analysis with attention-based interventions (Che et al., 2026). Existing benchmarks and datasets cover complementary settings, including natural-image counting in TallyQA (Acharya et al., 2019), controlled and synthetic numerosity settings such as SITUATE (Peinl et al., 2026) and NumerosityVLM (Fu et al., 2026), and visual quantity comparison in CompareBench (Cai et al., 2025). Our work instead jointly organizes training and evaluation along visual configuration, cardinality, and task interface, supported by controlled data, matched-budget studies, and internal readout analyses.

3 Structured Post-Training for Numerosity

3.1 Problem Formulation

Given an image x and a numerical query q, a vision-language model parameterized by θ defines a distribution $p_\theta(y \mid x, q)$ over answer sequences y and generates an answer $\hat{y}_\theta(x, q)$. The reference answer $y^\star$ may specify a cardinality, a comparison outcome, or the size of a query-defined set. We study *functional numerosity access*: the ability to produce the correct numerical response across visual configurations c, task-relevant cardinalities n, and task interfaces τ (Figure 1).

For an evaluation distribution $\mathcal{D}$ of images, queries, reference answers, and task interfaces, we measure *behavioral access* through exact-answer accuracy:

$$A_\theta(\mathcal{D}) = \mathbb{E}_{(x,q,y^\star,\tau)\sim\mathcal{D}}\left[\mathbf{1}\{\nu_\tau(\hat{y}_\theta(x,q)) = \nu_\tau(y^\star)\}\right], \tag{1}$$

where ν_τ normalizes generated and reference answers into a common task-specific format. Evaluating this accuracy across configurations, count ranges, and interfaces characterizes the reliability and transfer of numerical behavior.

Following the distinction between hidden quantity and symbolic output (Pang et al., 2026), we separately assess *internal quantity readout*: how accurately cardinality can be recovered from hidden representations using probes and count-specific prototypes.

Our learning problem is to adapt a pretrained model θ_0 to improve functional numerosity access. We organize supervision progressively along the three dimensions above. Configuration Coverage first stabilizes cardinality across complementary visual realizations; Relational Numerosity Supervision (RNS) connects and extends cardinalities through local changes and part–whole relations; and multi-interface supervision develops the use of quantity under different task demands.

We implement this design in two optimization phases. Coverage first initializes the model over $\mathcal{N}_{\text{cov}} = \{1, \ldots, K\}$, where K is the maximum cardinality supervised during initialization. A joint RNS and multi-interface update then develops numerical relations and task-conditioned operations from this checkpoint.

3.2 Controlled Numerosity Data

To study how the organization of supervision shapes numerical capability, we construct controlled synthetic training data and an evaluation suite spanning visual configurations, count ranges, and numerical queries.

Images are rendered from explicit object configurations, with reference answers derived from object counts and task rules. Our synthetic training data comprise Coverage examples at counts 1–8 and RNS examples with direct-count and relational answer targets up to 12. Synthetic evaluations assess local counting, robustness across visual conditions, and transfer to a separate 9–16 image distribution beyond the Coverage initialization range. Multi-interface supervision additionally uses task-specific subsets of existing CLEVR (Johnson et al., 2017) and TallyQA (Acharya et al., 2019) data, retaining their original images and annotations. Appendix A.1 documents construction, sampling, and content-overlap checks; separate representation-diagnostic datasets are described in Appendix A.4.

3.3 Cardinality-Conditioned Configuration Coverage

Coverage hypothesis. Configuration-based accounts of subitizing associate rapid enumeration with familiar spatial patterns, and learning canonical configurations can facilitate accurate enumeration of larger sets (Mandler & Shebo, 1982). This motivates our hypothesis: *pairing complementary visual configurations with shared cardinality labels can stabilize numerical access within the supervised cardinality range.*

We combine two sources of direct-count supervision. *Enumerative Coverage* systematically constructs occupancy patterns on a 4×4 lattice; *Factorized Coverage* varies spatial layouts, object properties, and backgrounds. *Combined Coverage* allocates equal numbers of examples to these sources, associating each cardinality with complementary visual realizations. Construction and sampling details appear in Appendix A.1.

We set the Coverage limit to $K = 8$, targeting an initial expansion beyond the approximately 1–4 range in which the pretrained model counts relatively reliably. This choice balances range expansion against the combinatorial growth of the enumerative configuration pool.

Let $p_{\text{cov}}(n, x)$ denote the empirical distribution of the combined corpus. We optimize

$$\mathcal{L}_{\text{cov}}(\theta) = \sum_{n=1}^{K} p_{\text{cov}}(n) \mathbb{E}_{x \sim p_{\text{cov}}(x|n)} \left[\ell_\theta(x, q_{\text{count}}, \text{str}(n)) \right], \tag{2}$$

where $p_{\text{cov}}(n)$ is the cardinality frequency, $p_{\text{cov}}(x \mid n)$ is the conditional image distribution, q_{count} is a direct-count query, and $\text{str}(n)$ converts the count into its answer string. The autoregressive answer loss ℓ_θ is defined in Appendix A.2. Shared targets encourage consistent numerical responses across the visual realizations represented at each cardinality.

3.4 Relational Numerosity Supervision

Relational hypothesis. Groupitizing motivates enumerating larger sets through smaller subgroups and their part–whole relationships (Wege et al., 2022). We hypothesize that *connecting cardinalities through local changes and part–whole relations can support accurate counting beyond the initial Coverage range.*

Relational Numerosity Supervision (RNS) combines direct-count retention and extension examples with relational examples. Direct-count supervision supplies absolute numerical targets, while relational supervision connects these targets through changes and composition. The query specifies the requested operation, with representative numerical targets

$$a^\star = \begin{cases} n, & \text{direct count,} \\ n+\delta, & \text{count after a change,} \\ n_b - n_a, & \text{change between two counts,} \\ \displaystyle\sum_{j=1}^{J} n_j, & \text{total of disjoint groups.} \end{cases} \tag{3}$$

Here, n is the directly counted or initial cardinality, δ is a signed addition or removal, n_a and n_b are the before- and after-change cardinalities, and n_j is the cardinality of group j among J disjoint groups. Each target is encoded as the reference answer $y^\star = \mathrm{str}(a^\star)$. RNS uses the same autoregressive answer objective as Coverage; the numerical relations determine the query and target rather than introducing a separate consistency loss. Appendix A.1 gives the instantiated task types and target ranges.

3.5 Multi-Interface Supervision

Interface hypothesis. *Supervision across numerical task interfaces can develop the reuse of quantity under different task demands, including comparison, compositional counting, and query-conditioned selection.*

Multi-interface supervision specifies both which objects are relevant and how their quantities should be used. For an image x and query q with task interface τ, let $\mathcal{S}_{\tau,1}, \ldots, \mathcal{S}_{\tau,J}$ denote the J relevant object sets. The reference answer is

$$y^\star_\tau = \mathrm{enc}_\tau(g_\tau(|\mathcal{S}_{\tau,1}|, \ldots, |\mathcal{S}_{\tau,J}|)), \tag{4}$$

where $|\mathcal{S}_{\tau,j}|$ is the cardinality of set j, g_τ returns the requested numerical result, and enc_τ converts that result into the reference answer string. Filtering, intersection, and union determine the relevant sets before their cardinalities are evaluated.

We instantiate this supervision with three sources derived from CLEVR (Johnson et al., 2017) and TallyQA (Acharya et al., 2019). CLEVR Num supplies equality and greater-/less-than comparisons; CLEVR Comp supplies filtered and compositional counting; and TallyQA supplies query-conditioned counting over natural images. Their mean supervised answer losses are denoted by $\mathcal{L}_{\mathrm{num}}$, $\mathcal{L}_{\mathrm{comp}}$, and $\mathcal{L}_{\mathrm{tally}}$, respectively.

3.6 Two-Phase Optimization

Coverage initializes configuration-robust counting, after which joint RNS and multi-interface training develops cardinality extension and task-conditioned reuse (Figure 2).

The second phase combines RNS, numerical comparison, composition, and TallyQA supervision:

$$\mathcal{L}_{\mathrm{mix}}(\theta) = \lambda_R \mathcal{L}_{\mathrm{rel}}(\theta) + \lambda_N \mathcal{L}_{\mathrm{num}}(\theta) + \lambda_C \mathcal{L}_{\mathrm{comp}}(\theta) + \lambda_T \mathcal{L}_{\mathrm{tally}}(\theta). \tag{5}$$

Each $\mathcal{L}$ is the mean answer loss on its source corpus. The coefficients λ_R, λ_N, λ_C, and λ_T are the corresponding corpus fractions, satisfying $\lambda_t \geq 0$ and $\sum_{t\in\{R,N,C,T\}} \lambda_t = 1$.

Let $\mathrm{FT}(\theta; \mathcal{L})$ denote supervised fine-tuning initialized at θ. Starting from pretrained parameters θ_0, the two phases produce

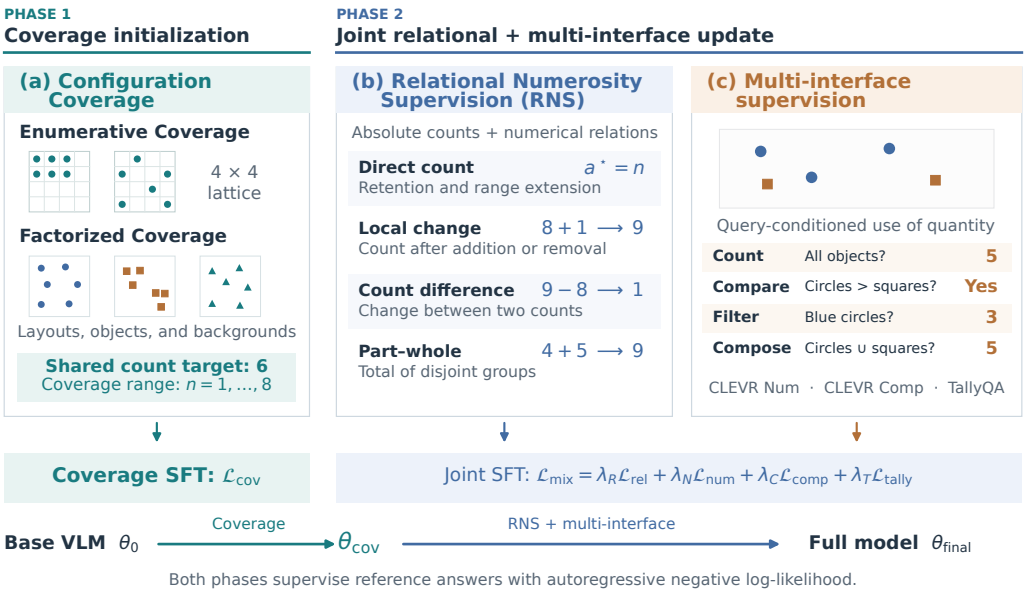


Figure 2: **Two-phase post-training framework.** Coverage initializes counting across complementary visual configurations at counts 1–8; a joint RNS and multi-interface update then develops cardinality extension and task-conditioned numerical use.

$$\theta_{\text{cov}} = \text{FT}(\theta_0; \mathcal{L}_{\text{cov}}), \qquad \theta_{\text{final}} = \text{FT}(\theta_{\text{cov}}; \mathcal{L}_{\text{mix}}). \tag{6}$$

Both phases optimize low-rank adapters. Exact source allocations, training budgets, and optimization settings are reported in Appendix A.2.

4 Experiments

We evaluate overall numerical performance across two VLM families, analyze supervision design through matched-budget studies, and examine internal numerical readout.

4.1 Experimental Setup

Models and training configurations. We evaluate Qwen3-VL-8B-Instruct as the primary model and InternVL3-9B-Instruct for cross-family replication. Alongside the unmodified **Base VLM**, we compare the three configurations in Table 1. Each uses 4,096 downstream examples and 512 updates; Full additionally receives Coverage initialization. Source allocations and optimization details appear in Appendix A.2.

Configuration	Coverage init.	RNS	Interface supervision
Interface-only	No	No	Yes
RNS+Interface	No	Yes	Yes
Full	Yes	Yes	Yes

Table 1: **Main training configurations.** RNS replaces part of the interface supervision at a fixed downstream sample budget. RNS+Interface and Full use the same downstream mixture.

Evaluation. Local synthetic counting measures accuracy within the Coverage range (1–8) and higher-count acquisition (9–12). Synthetic OOD 9–16 tests transfer to a separately constructed visual distribution. Counts 9–12 are also included in downstream RNS supervision, so this is not strict unseen-count extrapolation. PixMo-Count (accessible test subset) (Deitke et al., 2025) and CompareBench Quantity (Cai et al., 2025) test external natural-image counting and quantity comparison, respectively, with neither benchmark used for supervision. Held-out CLEVR and

TallyQA evaluations assess retention on supervised task families. Both backbones use the same evaluation manifests. We report exact-answer accuracy. Synthetic data construction and content checks appear in Appendix A.1, with decoding and statistical protocols in Appendix A.2.

4.2 Overall Numerical Performance

High-precision counting within and beyond the Coverage range. The complete framework achieves 99.83% accuracy at counts 1–8 and 85.42% at counts 9–12 on our local synthetic evaluation (Table 2). These results establish high-precision counting within the Coverage range and extend accurate counting to larger cardinalities through downstream supervision.

	Coverage range		Larger counts
Training	1–4	5–8	9–12
Interface-only	97.87	57.48	27.78
RNS+Interface	100.00	93.54	74.65
Full	100.00	99.66	85.42

Table 2: **Local counting capability on Qwen.** Exact accuracy (%), averaged over three downstream seeds. Counts 1–8 lie within Coverage's range; counts 9–12 are additionally supervised during downstream RNS training. Interface-only and RNS+Interface use the same downstream sample budget; Full uses the same downstream mixture as RNS+Interface with additional Coverage initialization.

Generalization across visual distributions and model families. Full improves Synthetic OOD 9–16 accuracy from 30.00% to 49.42% on Qwen and from 14.25% to 54.13% on InternVL relative to their pretrained baselines (Table 3). Both backbones also improve on PixMo-Count and CompareBench Quantity, although the gains vary across benchmarks and model families. The strongest improvements occur on held-out synthetic distributions, with more modest gains on external numerical tasks.

Training	Synthetic OOD 9–16	PixMo-Count	CompareBench Quantity
Qwen3-VL-8B-Instruct			
Base VLM	30.00	65.52	61.00
Interface-only	34.96	69.59	60.78
RNS+Interface	39.13	69.33	59.78
Full	**49.42**	**70.73**	**61.67**
InternVL3-9B-Instruct			
Base VLM	14.25	63.81	54.33
Interface-only	20.75	64.19	54.17
RNS+Interface	28.75	62.48	54.67
Full	**54.13**	**64.38**	**55.17**

Table 3: **Transfer to held-out evaluations across two VLM backbones.** Exact accuracy (%). Qwen post-training results are three-seed means; InternVL uses one run. Synthetic OOD 9–16, PixMo-Count, and CompareBench Quantity are excluded from the corresponding downstream training mixtures. Base VLM denotes the unmodified checkpoint; its evaluation protocol is given in Appendix A.2.

Retention of supervised numerical tasks. Full retains most of the held-out CLEVR and TallyQA performance of Interface-only while allocating less direct supervision to these task families (Table 4). The framework therefore develops stronger counting capability alongside largely preserved task-specific performance.

4.3 Ablation and Supervision Design

Coverage initialization and complementary configurations. With the downstream mixture held fixed, Full outperforms RNS+Interface on Synthetic OOD 9–16 by 10.29 percentage points on Qwen and 25.38 points on InternVL (Table 3), demonstrating a persistent benefit from Coverage initialization. Within the Coverage phase, matched-budget comparisons further distinguish the

	Downstream training allocation				Held-out performance	
Training	RNS	Num	Comp	TallyQA	CLEVR	TallyQA
Base VLM	0	0	0	0	80.75	79.80
Interface-only	0	1024	1024	2048	93.57	80.53
Full	1638	614	614	1230	91.35	80.00

Table 4: **Downstream supervision reallocation and task retention on Qwen.** The two post-training conditions use 4,096 downstream examples; Base VLM receives none. Full reallocates part of the interface-supervision budget to RNS while retaining most held-out task performance. Num and Comp denote CLEVR quantity-comparison and compositional-counting supervision.

effects of supervision composition: Combined Coverage achieves 96.42% mean accuracy across visual conditions, compared with 91.77% for Enumerative Coverage and 93.41% for Factorized Coverage (Table 5a). The same ordering holds on larger grids. Thus, combining systematic occupancy patterns with varied layouts and appearances improves robustness at a fixed sample budget. Full condition-level results appear in Appendix A.3.1.

(a) Configuration coverage

Coverage	Visual conditions (1–8)	Larger grid
Enumerative	91.77	81.25
Factorized	93.41	86.75
Combined	**96.42**	**92.75**

(b) Multi-interface supervision

Supervision	Comp.	Comp. (5–8)	Quantity
RNS	52.10	14.90	93.20
RNS+Num	72.50	51.63	95.10
RNS+Num+Comp	**88.30**	**83.27**	**96.50**

Table 5: **Matched-budget supervision-design studies on Qwen.** Exact accuracy (%). Panel (a) compares alternative Coverage compositions at the same total training-example budget. Panel (b) starts from the same Coverage checkpoint and reallocates a fixed downstream budget among relational and task-interface supervision. Num and Comp denote CLEVR quantity-comparison and compositional-counting supervision.

Relational supervision supports higher-count acquisition. Starting from the same Base VLM, reallocating part of the fixed downstream budget from interface supervision to RNS raises local accuracy at counts 9–12 from 27.78% to 74.65% on Qwen (Table 2). Its improvement on Synthetic OOD is smaller (Table 3), indicating that this allocation primarily strengthens higher-count acquisition. This comparison evaluates the contribution of the RNS corpus, which combines direct-count extension with relational examples.

Interface allocation develops task-specific numerical use. From a shared Coverage checkpoint, reallocating a fixed downstream budget to quantity comparison and then compositional counting increases composition accuracy from 52.10% to 72.50% and 88.30%, respectively (Table 5b). Gains are particularly pronounced at counts 5–8. These results show that explicit supervision of numerical interfaces develops task-specific uses of the counting capability established during initialization.

4.4 Internal Numerical Readout

We ask whether post-training makes quantity more accurately decodable from intermediate representations and whether a count readout transfers across visual configurations. We freeze four Qwen checkpoints—Base VLM, Coverage, RNS+Interface, and Full—and extract $h_\theta(x, q)$, the language-block-17 representation at the last prompt token before answer generation, for image x and counting query q. We use two representation diagnostics and separately evaluate generated answers.

Quantity regression. Following the use of independent linear probes to assess intermediate representations (Alain & Bengio, 2016), we fit a ridge regressor $\widehat{n}_{\text{reg}} = w^\top \widetilde{h}_\theta(x, q) + b$ for each checkpoint. Here $\widetilde{h}_\theta$ is standardized using training-set statistics, and w, b are fitted readout parameters. Probes are trained on counts 1–8 and evaluated by mean absolute error (MAE) at 1–8, 9–12, and 13–16. The latter ranges test extrapolation of the fitted readout beyond its training range.

	Middle-layer probe MAE (↓)			Cross-layout accuracy (1–8; %, ↑)	
Checkpoint	1–8	9–12	13–16	Prototype	Generation
Base VLM	0.140	0.946	3.258	65.31	86.33
Coverage	0.071	0.627	2.478	85.00	64.53
RNS+Interface	0.113	0.853	3.122	59.84	88.12
Full	0.073	0.628	2.501	81.33	86.33

Table 6: **Internal numerical readout and generated answers on Qwen.** Left: block-17 ridge-probe MAE, with fitting restricted to counts 1–8. Right: fixed Grid-prototype classification and generated-count accuracy on the same Scattered and Clustered target images. The two column groups use separate diagnostic datasets. Results are point estimates for fixed checkpoints.

Cross-configuration prototype transfer. On a separate diagnostic dataset at counts 1–8, we construct count-specific prototypes from Grid reference images and classify Scattered and Clustered images by cosine similarity:

$$\mu_{\theta,a,n} = \frac{1}{|\mathcal{R}_{a,n}|} \sum_{x \in \mathcal{R}_{a,n}} h_\theta(x, q), \qquad \widehat{n}_{\text{proto}}(x; a) = \operatorname*{arg\,max}_{k \in \{1,\ldots,8\}} \cos(h_\theta(x, q), \mu_{\theta,a,k}) \,. \tag{7}$$

Here $\mathcal{R}_{a,n}$ contains Grid reference images of count n under size condition a: fixed object radius or approximately fixed total area. Prototypes are constructed separately for each checkpoint and size condition, then held fixed across target layouts. We also evaluate exact generated-count accuracy on these same target images using the behavioral metric in Eq. 1.

Coverage improves quantity readout and its cross-configuration transfer. Coverage and Full have lower probe MAE than Base VLM and RNS+Interface in all three count ranges (Table 6). Coverage also raises cross-layout prototype accuracy from 65.31% to 85.00%; Full retains 81.33% after joint training. These results identify a Coverage-associated improvement in both intermediate quantity decodability and the transfer of count-specific readouts across layouts.

Joint training retains readout gains while recovering generation. On the prototype-target images, generation accuracy falls from 86.33% for Base VLM to 64.53% after Coverage, then recovers to 86.33% after joint training. Full therefore matches Base generation accuracy while retaining stronger prototype transfer. This training trajectory connects to the distinction between readable quantity and symbolic output (Pang et al., 2026).

Readout controls. Shuffling reference count labels reduces prototype accuracy to approximately 12.5%, the eight-way chance level, establishing dependence on correctly labeled references. A separate image-statistics control, including renderer-derived metadata, reaches 94.69%, identifying strong low-level cues correlated with cardinality in the stimuli. Full protocol and control details are provided in Appendix A.4.2.

5 Discussion and Conclusion

Our results highlight supervision organization as an important factor in developing visual numerosity. At a fixed sample budget, complementary configuration coverage improves robustness, while relational and interface-specific mixtures support higher-count and task-conditioned performance. On Qwen, representation analyses further reveal more accurate and reusable quantity readout.

We characterize this combination of precision and selective reuse as *numerosity specialization*. The framework develops high-precision local counting and improves synthetic-distribution transfer across two VLM families, while gains on external numerical tasks remain more variable. These results suggest that strengthening visual number sense requires not only extending counting accuracy, but also organizing supervision so that quantity can remain accessible across configurations and task interfaces.

AI USE STATEMENT

Generative AI tools were used to assist with language polishing, grammar and consistency checks, and the preparation of conceptual illustrations. All scientific claims, experimental results, analyses, and conclusions were produced and verified by the authors, who take full responsibility for the content of this paper.

ETHICS STATEMENT

This work does not involve human subjects, personal data, or deployment decisions. The experiments use synthetic data generated by the authors and publicly available datasets and benchmarks. We are not aware of any additional ethical concerns specific to this study.

REPRODUCIBILITY STATEMENT

We provide the experimental settings required to reproduce our results, including model configurations, training data composition, random seeds, optimization settings, evaluation protocols, prompts, and metrics. Upon acceptance, we will publicly release the constructed datasets, training and evaluation code, and trained model checkpoints to facilitate full reproduction and further research.

REFERENCES

Manoj Acharya, Kushal Kafle, and Christopher Kanan. Tallyqa: Answering complex counting questions. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 8076–8084, 2019.

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

Giovanni Anobile, Elisa Castaldi, Paula A Maldonado Moscoso, David C Burr, and Roberto Arrighi. "groupitizing": a strategy for numerosity estimation. *Scientific Reports*, 10(1):13436, 2020.

Jie Cai, Kangning Yang, Lan Fu, Jiaming Ding, Jinlong Li, Huiming Sun, Daitao Xing, Jinglin Shen, and Zibo Meng. Comparebench: A benchmark for visual comparison reasoning in vision-language models. *arXiv preprint arXiv:2509.22737*, 2025.

Declan Campbell, Sunayana Rane, Tyler Giallanza, Nicolò De Sabbata, Kia Ghods, Amogh Joshi, Alexander Ku, Steven M Frankland, Thomas L Griffiths, Jonathan D Cohen, et al. Understanding the limits of vision language models through the lens of the binding problem. *Advances in Neural Information Processing Systems*, 37:113436–113460, 2024.

Liwei Che, Zhiyu Xue, Yihao Quan, Benlin Liu, Zeru Shi, Michelle Hurst, Jacob Feldman, Ruixiang Tang, Ranjay Krishna, and Vladimir Pavlovic. Counting circuits: Mechanistic interpretability of visual reasoning in large vision-language models. *arXiv preprint arXiv:2603.18523*, 2026.

Samuel J Cheyette and Steven T Piantadosi. A unified account of numerosity perception. *Nature human behaviour*, 4(12):1265–1272, 2020.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 91–104. IEEE, 2025.

Lisa Feigenson, Stanislas Dehaene, and Elizabeth Spelke. Core systems of number. *Trends in cognitive sciences*, 8(7):307–314, 2004.

Yiming Fu, Fangjun Li, Xiujin Liu, Ruidong Ma, Hang Yu, Zhichen Lu, Kanwei He, Alessandro Di Nuovo, Angelo Cangelosi, and Zhegong Shangguan. Numerosityvlm: A cognitively inspired benchmark for interpreting numerosity representations in vision-language models. *arXiv preprint arXiv:2608.15425*, 2026.

Hosein Hasani, Amirmohammad Izadi, Fatemeh Askari, Mobin Bagherian, Sadegh Mohammadian, Mohammad Izadi, and Mahdieh Soleymani Baghshah. Understanding counting mechanisms in large language and vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5125–5133, 2026.

Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2901–2910, 2017.

George Mandler and Billie J Shebo. Subitizing: an analysis of its component processes. *Journal of experimental psychology: general*, 111(1):1, 1982.

Roni Paiss, Ariel Ephrat, Omer Tov, Shiran Zada, Inbar Mosseri, Michal Irani, and Tali Dekel. Teaching clip to count to ten. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3147–3157. IEEE, 2023.

Xingzhou Pang, Yifan Hou, Junling Wang, and Mrinmaya Sachan. Unveiling the visual counting bottleneck in vision-language models. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, Seoul, South Korea, 2026. URL https://arxiv.org/abs/2605.30170.

René Peinl, Vincent Tischler, Patrick Schröder, and Christian Groth. Situate–synthetic object counting dataset for vlm training. *arXiv preprint arXiv:2602.00108*, 2026.

Manuela Piazza, Philippe Pinel, Denis Le Bihan, and Stanislas Dehaene. A magnitude code common to numerosities and number symbols in human intraparietal cortex. *Neuron*, 53(2):293–305, 2007.

Huajie Tan, Yuheng Ji, Xiaoshuai Hao, Xiansheng Chen, Pengwei Wang, Zhongyuan Wang, and Shanghang Zhang. Reason-rft: Reinforcement fine-tuning for visual reasoning of vision language models. *Advances in neural information processing systems*, 38:5772–5822, 2026.

Lana M Trick and Zenon W Pylyshyn. Why are small and large numbers enumerated differently? a limited-capacity preattentive stage in vision. *Psychological review*, 101(1):80, 1994.

Theresa E Wege, Kelly Trezise, and Matthew Inglis. Finding the subitizing in groupitizing: Evidence for parallel subitizing of dots and groups in grouped arrays. *Psychonomic Bulletin & Review*, 29 (2):476–484, 2022.

A Additional Experimental Details

A.1 Controlled Numerosity Data

We generate synthetic images, queries, and reference answers from object coordinates and task rules. Our data resource includes Coverage and RNS training data, behavioral evaluations, and separate representation diagnostics. CLEVR and natural-image data retain their original images and annotations; our contribution for these sources is task selection and evaluation protocols.

Configuration Coverage. The enumerative pool contains all $\sum_{n=1}^{8} \binom{16}{n} = 39{,}202$ occupancy patterns with 1–8 objects on a fixed 4×4 lattice. Images are 224×224, with black circles of radius 10 pixels on white backgrounds; lattice centers span normalized coordinates 0.14–0.86. The factorized pool contains another 39,202 examples across nine layouts: sparse random, dense random, linear, ring, rectangular grid, dice, ten-frame, proximity clusters, and common-region clusters. It varies four shapes, six foreground colors, five backgrounds, and three textures, with object radius 0.027–0.049 of image width. Spatial transformations and occasional partial occlusion add variation. Labels count constructed objects; factorized rendering does not enforce collision-free or fully visible objects.

Combined Coverage uses 19,601 enumerative examples sampled without replacement (seed 20260728) and the first 19,601 factorized records in generation order, followed by shuffling. Its count frequencies for 1–8 are 2,460, 2,511, 2,727, 3,363, 4,613, 6,450, 8,211, and 8,867. All three corpora use the query *How many target objects are in the image? Return only one Arabic numeral.*

Relational examples and task interfaces. RNS uses single 224×224 images or paired 466×224 images with an 18-pixel gap. Addition and successor examples preserve the left-panel coordinates within the larger right-panel set; removal preserves a subset of the initial coordinates. Part–whole examples render two disjoint groups. Table 7 distinguishes displayed counts from the answer requested by each query. CLEVR functional programs determine relevant object sets and deduplicate unions and intersections by object identity. Official training images provide supervision and validation images provide evaluation. The final TallyQA training subset combines 615 complex and 615 imported-source simple-proxy questions. Source allocations appear in Table 8.

Task	Examples	Scene construction	Answer
Direct retention	480	One set of 1–8 objects	Set size
Direct extension	241	One set of 9–12 objects	Set size
Count difference	253	Left: 8; right: 9–12	Difference, 1–4
Anchor to total	251	Left: 8; right: 9–12	Right-panel size
Removal	160	Initial: 10–16; remove 1–4; retain 9–12	Remaining size
Partition and sum	168	Two sets, each 3–9; total 9–12	Sum of set sizes
Successor	85	Left: 8–11; right: one more	Right-panel size

Table 7: **Final RNS composition.** All 1,638 examples enter the joint update. Reference answers lie in 1–12, although individual panels can contain up to 16 objects.

Synthetic evaluation distributions. The within-range robustness suite contains 9,600 questions: 100 per count from 1–8 in each of 12 conditions. These vary color, shape, size, background, jitter, rotation, texture, larger-grid layout, continuous positions, nonuniform spacing, clustering, and irregular arrangement. Default images contain black circles of radius 8 pixels on white 224×224 canvases. Some condition families also appear in factorized training, and clustered or irregular scenes may contain touching objects.

Synthetic OOD 9–16 contains 800 distinct images, 100 per count. Centers are sampled in $[0.08, 0.92]^2$ with minimum separation 0.075; circles have radius 8 pixels. This extends the Coverage range and changes the enumerative grid distribution. RNS directly supervises 9–12, while larger panel counts and natural-image answers also occur downstream, so the full 9–16 interval is not unseen throughout training. Both synthetic suites use generation seed 20260722 and the Coverage counting query.

The internal 640-question manifest spans seven RNS task types. Selecting its two direct-count types gives 288 questions: 192 at counts 1–8 and 96 at 9–12, with 283 distinct image-file hashes. This is

the Local direct-count subset documented in the dataset audit below. Representation-diagnostic data and their layout-grouped splits are described separately in Appendix A.4.

Train–evaluation overlap audit. A file-byte SHA256 audit across all training manifests (Coverage and downstream supervision) finds no identical training images in the local direct-count, visual-robustness, Synthetic OOD, CLEVR, TallyQA test, PixMo-Count, or CompareBench evaluation sets.

A.2 Training and Evaluation Protocols

Training configurations. **Interface-only** starts from the pretrained model and uses its full downstream budget for CLEVR and TallyQA interface supervision. **RNS+Interface** starts from the same base model but reallocates part of that budget to RNS. **Full** uses the identical downstream mixture after Coverage initialization. Each downstream run uses 4,096 examples and 512 updates; Full additionally receives the Coverage phase. Table 8 gives the allocations for the main comparisons and interface-design studies.

Configuration	Initialization	RNS	Num	Comp	TallyQA
Interface-only	Base	0	1024	1024	2048
RNS+Interface	Base	1638	614	614	1230
Full	Coverage	1638	614	614	1230
RNS	Coverage	4096	0	0	0
RNS+Num	Coverage	3072	1024	0	0
RNS+Num+Comp	Coverage	2048	1024	1024	0

Table 8: **Downstream training-example allocations.** Each row contains 4,096 examples. Interface-only, RNS+Interface, and Full are the main comparison configurations. RNS+Interface and Full use the identical downstream mixture but differ in Coverage initialization. The three interface-design conditions in the lower block share the same Coverage checkpoint.

Optimization. Coverage uses 39,202 examples. For Qwen3-VL-8B-Instruct, Coverage is trained for one epoch with effective batch size 16 and learning rate 2×10^{-4}. Downstream runs use 512 updates, effective batch size eight, and learning rate 10^{-4}. Qwen uses 4-bit NF4 LoRA with rank 16, alpha 32, dropout 0.05, and adapters on the q/k/v/o/gate/up/down projections. Main downstream comparisons use three seeds: 20260813, 20260818, and 20260819.

InternVL3-9B-Instruct uses the same 39,202-example Coverage budget and 512-update downstream budget. Its Coverage and downstream learning rates are 2×10^{-4} and 10^{-4}, respectively. Language-layer LoRA uses rank 16, alpha 32, dropout 0.05, and wqkv/wo/w1/w2/w3 projections. InternVL results use seed 20260813.

Answer supervision. For a reference sequence $y^\star = (y_1^\star, \ldots, y_L^\star)$, all supervision uses the autoregressive objective

$$\ell_\theta(x, q, y^\star) = -\sum_{j=1}^{L} \log p_\theta(y_j^\star \mid x, q, y_{<j}^\star). \tag{8}$$

Relations and task interfaces determine the query and reference answer; no additional consistency objective is used. In the joint downstream mixture, source weights correspond to corpus fractions.

Task-family and external evaluations. CLEVR composition and quantity comparison each contain 1,000 questions. The main CLEVR result averages these two task families. TallyQA contains 500 Simple and 500 Complex questions and is likewise averaged across the two subsets. CompareBench Quantity contains 600 four-image quantity-comparison questions.

PixMo-Count is excluded from all training mixtures. All checkpoints are evaluated on the same 525 accessible test images at counts 2–10 using the prompt "How many [OBJECT] are there in the image? Answer with a single number." Decoding is greedy.

Qwen Base evaluation. The unmodified Qwen3-VL-8B-Instruct checkpoint is evaluated without an adapter. On Synthetic OOD, Base achieves 30.00% exact accuracy (32.25% at counts 9–12 and 27.75% at counts 13–16). Base reaches 61.00% on CompareBench Quantity. CLEVR quantity comparison and filtered/compositional counting score 91.60% and 69.90%, respectively, giving an equal-weight mean of 80.75%. TallyQA Simple and Complex score 85.20% and 74.40%, giving a mean of 79.80% on the evaluation subsets described above. Synthetic OOD and CLEVR use batch size 32; CompareBench and TallyQA use batch size eight. These evaluations use greedy decoding with at most four new tokens.

Metrics and uncertainty. Exact accuracy compares normalized responses with references; invalid outputs are incorrect. Synthetic behavioral scoring extracts the first signed integer and uses all questions as its denominator. Qwen downstream results are reported over three training seeds using sample standard deviation. Paired bootstrap intervals use 10,000 resamples and condition on the evaluated checkpoints. InternVL reports one training seed; uncertainty therefore reflects paired evaluation-sample variation rather than training-seed variability.

A.3 Additional Supervision-Design Results

A.3.1 Configuration Coverage

The three Coverage conditions use the source corpora described in Appendix A.1. Their total training-example budget is fixed at 39,202, while count frequencies and visual configurations vary. Table 9 reports the condition-level robustness results underlying the main-paper averages.

Condition	Base	Enumerative	Factorized	Combined
Color	84.25	100.00	96.37	100.00
Shape	79.13	89.12	95.25	92.87
Size	80.75	99.88	92.63	99.88
Background	83.63	100.00	99.00	100.00
Jitter	82.13	97.50	94.50	99.62
Rotation	76.00	98.88	95.13	99.62
Texture	78.75	99.88	99.25	99.88
Larger grid	80.50	81.25	86.75	92.75
Continuous	83.50	86.25	90.75	94.75
Nonuniform	80.63	81.87	87.62	92.50
Clustered	84.13	83.75	93.63	93.87
Irregular	83.25	82.87	90.00	91.25
Average	81.39	91.77	93.41	96.42

Table 9: **Coverage design study.** Exact accuracy (%) at counts 1–8. Single-run comparisons use the same total training-example budget; the average equally weights all 12 conditions.

A.3.2 Relational and Interface Allocation

The main relational comparison contrasts Interface-only with RNS+Interface. Both start from the Base VLM and use the same 4,096-example downstream budget, while RNS+Interface reallocates 1,638 examples from interface supervision to RNS. This comparison therefore measures the effect of introducing relational numerosity supervision through budget reallocation, rather than increasing the amount of downstream supervision.

For the interface-design study, RNS, RNS+Num, and RNS+Num+Comp share the same Coverage checkpoint, training-example budget, and update budget. Examples are progressively reallocated from RNS to numerical comparison and compositional counting, allowing the effect of task interface allocation to be examined at fixed downstream scale.

A separate natural-image allocation comparison adds TallyQA supervision while preserving the 4,096-example downstream budget. Its task-level results are shown in Table 10.

Supervision	TallyQA Simple	TallyQA Complex	CLEVR Comp
RNS+Num+Comp	84.20 ± 0.72	69.53 ± 0.31	87.57 ± 0.83
Full	85.73 ± 0.64	74.27 ± 0.83	87.10 ± 1.82

Table 10: **Natural-image supervision allocation.** Exact accuracy (%), mean $\pm$ sample SD over three downstream seeds. Both mixtures start from the same Coverage checkpoint and contain 4,096 downstream examples. RNS+Num+Comp contains no TallyQA supervision, whereas Full reallocates part of the same downstream budget to TallyQA.

A.4 Internal Numerical Readout

A.4.1 Ridge-Probe Protocol

The regression diagnostic contains 1,920 nonoverlapping blue-circle images at counts 1–16. Images are 448×448 and vary two backgrounds and two size rules: fixed object radius and approximately fixed total object area. Layout-grouped training, validation, and test splits contain 1,152, 384, and 384 images. Ridge fitting and validation use only counts 1–8; counts 9–12 and 13–16 evaluate readout extrapolation. These fitting splits train readouts, not the VLM.

For each checkpoint, hidden states are extracted at the last prompt token before answer generation. The main paper reports language block 17. Features are standardized using training-set statistics, and validation MAE selects the ridge penalty from $\{0.1, 1, 10, 100, 1000\}$. Probes are fitted independently for each checkpoint, and predictions are not rounded or clipped.

A.4.2 Cross-Configuration Prototypes

The prototype diagnostic contains 5,760 images crossing counts 1–8, Grid/Scattered/Clustered layouts, two backgrounds, and the same two size rules used above. Disjoint scene families provide 2,880 reference, 960 development, and 1,920 test images from 30, 10, and 20 families, respectively. All count, configuration, and rendering variants of a family remain in one split. Hidden states are taken from language block 17 at the last valid prompt token before answer generation.

For checkpoint θ, source layout s, size rule a, and count n, the prototype is

$$\mu_{\theta,s,a,n} = \frac{1}{|\mathcal{R}_{s,a,n}|} \sum_{x \in \mathcal{R}_{s,a,n}} h_\theta(x, q), \qquad \widehat{n}_{\theta,s,a}(x) = \operatorname*{arg\,max}_{k \in \{1,\dots,8\}} \cos(h_\theta(x, q), \mu_{\theta,s,a,k}) \,. \tag{9}$$

The primary endpoint transfers Grid-derived prototypes to the 1,280 Scattered and Clustered test images without refitting. Generation is evaluated on the same target images. Prototypes are constructed independently for each checkpoint and size rule and remain fixed at test time.

Checkpoint	Prototype [CI]	Generation [CI]	Target reference	Gap (pp)
Base	65.31 [63.28,67.50]	86.33 [83.91,88.52]	95.86	30.55
Coverage	85.00 [83.36,86.72]	64.53 [62.81,66.33]	100.00	15.00
RNS+Interface	59.84 [57.97,62.03]	88.12 [86.25,89.92]	98.52	38.67
Full	81.33 [79.06,83.52]	86.33 [84.61,87.58]	100.00	18.67

Table 11: **Cross-configuration prototype transfer.** Prototype and generated-count accuracy (%) are measured on the same Scattered and Clustered target images. Target reference uses prototypes constructed directly from the target layouts; the gap is diagnostic rather than a theoretical upper bound.

Controls. Across 32 reference-label permutations, prototype accuracy is approximately 12.5%, the chance level for eight-way classification. We additionally fit an image-statistics ridge control using nine renderer-derived visual features: foreground occupancy ratio, image mean and standard deviation, mean absolute horizontal and vertical pixel differences, foreground bounding-box area, foreground density within the bounding box, minimum object-center spacing, and foreground-mask edge count. The control uses the same source-reference and target-test splits as the prototype analysis,

with feature standardization fitted on the source reference set and ridge regularization $\alpha = 1$. It reaches 94.69% accuracy on the primary Grid-to-Scattered/Clustered transfer, showing that the synthetic stimuli contain strong low-level cues correlated with cardinality. These controls therefore contextualize rather than establish cue-independent numerical representations.

A.4.3 Visualization of Cross-Configuration Features

We visualize the same 1,920 test images from the prototype diagnostic across the four fixed Qwen checkpoints. Each image contributes its 4,096-dimensional final-prompt-token representation at language block 17 (zero-based indexing). For each checkpoint, we apply ℓ_2 normalization, PCA to 50 dimensions, and an independent two-dimensional t-SNE fit with perplexity 30, PCA initialization, random seed 42, an automatically selected learning rate, and 1,500 iterations. The examples and settings are identical across checkpoints. Colors indicate counts 1–8, and marker shapes indicate Grid, Scattered, and Clustered layouts.

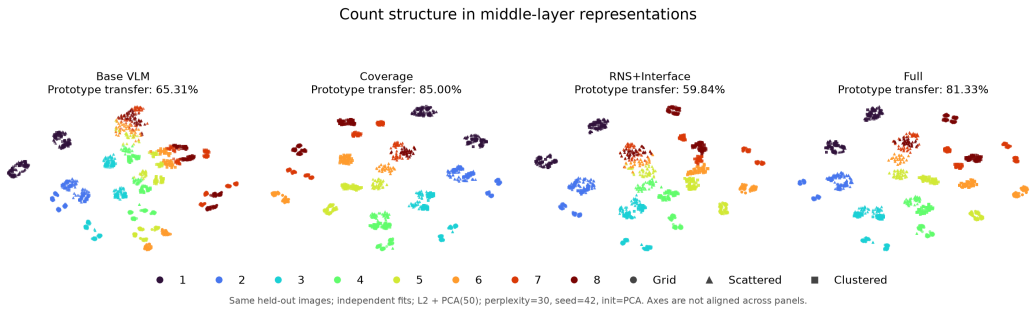


Figure 3: **Cross-configuration quantity representations.** t-SNE projections of intermediate-layer features from Base VLM, Coverage, RNS+Interface, and Full. Coverage and Full exhibit clearer grouping by cardinality, alongside remaining configuration-related subgroups. Each panel is fitted independently, so absolute positions and distances across panels are not comparable. The visualization complements the quantitative prototype-transfer results in Table 11.

The clearer count grouping after Coverage is largely retained after joint training, consistent with the stronger cross-configuration prototype readout. Alternative perplexities of 15 and 50 and a random-initialized seed-7 run preserve this broad qualitative pattern. Layout-related subgroups remain, and the visualization concerns only the 1–8 diagnostic range.