WHY MLLMS STRUGGLE TO COUNT: OVERCOMING INDIVIDUATION AND AGGREGATION BOTTLENECKS WITH CONVSTACK

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal Large Language Models (MLLMs) still struggle with fine-grained visual counting, especially as the number of objects increases. We study where these errors come from and identify two recurring bottlenecks. The first arises during object individuation: non-overlapping patchification can split a single object across multiple visual tokens, making it harder to form coherent object-level representations. The second appears during aggregation, where attention normalization can reduce the separation between representations of neighboring counts as numerosity grows. Based on these observations, we introduce ConvStack, a lightweight module that mixes local spatial information directly in the visual token space and injects the resulting features through zero-initialized residual connections. This design helps preserve local object structure while providing a more stable visual signal for downstream counting. Although ConvStack is trained only on counting data while updating about 0.5% of the model parameters, its gains extend beyond counting to broader spatial understanding, while general visual performance remains largely intact. Our mechanistic analysis shows that ConvStack improves the separation of counting representations and the encoding of spatial information within MLLMs.

1 INTRODUCTION

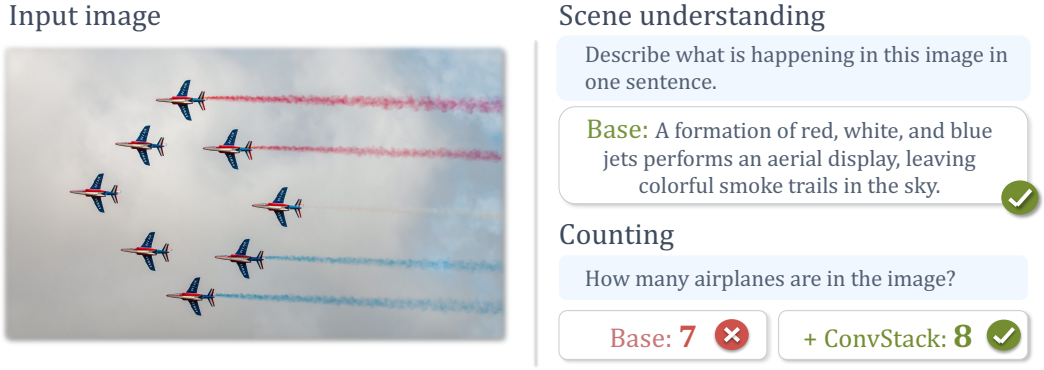


Figure 1: **Scene understanding versus counting.** Qwen3-VL-8B correctly describes the aerial display but counts seven airplanes instead of eight. ConvStack (ours) recovers the correct count.

Modern Multimodal Large Language Models (MLLMs) (Bai et al., 2025; Wang et al., 2025; Google DeepMind, 2024) inherit much of their visual capability from pretrained Vision Transformer (ViT) encoders (Dosovitskiy et al., 2021; Radford et al., 2021; Liu et al., 2023). While these encoders provide strong global representations that align well with language models, they remain surprisingly fragile on tasks requiring precise visual grounding. As shown in Figure 1, an MLLM may accurately describe the semantic contents of an image while simultaneously failing to correctly answer how many objects are present (Hasani et al., 2025; Che et al., 2026; Guo et al., 2025).

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.

We use visual counting as a minimal, highly revealing probe for this broader grounding deficit. Counting requires far more than merely detecting the presence of foreground semantics. A capable model must first parse raw patches into discrete, localized object units (*object individuation*) and then accumulate those units into a discrete numerosity (*counting aggregation*). In this work, we conduct a mechanistic analysis that identifies severe bottlenecks at both of these critical stages (Section 3). First, we reveal an *individuation bottleneck* stemming from image patchification. Because standard Vision Transformers project nonoverlapping patches independently, they severely struggle to group fragmented geometric features across patch boundaries, making it difficult to form coherent representations of individual objects. Second, we identify an *aggregation bottleneck* driven by attention normalization. Even when objects are successfully individuated, they must still be aggregated by the network's attention mechanisms. Under a homogeneous single-head model with fixed object and background representations, attention that favors objects yields progressively smaller signal increments between adjacent counts. This compression reduces the hidden state distances between neighboring counts and increases the sensitivity required for exact readout.

These twin diagnostic insights naturally dictate our architectural solution: a robust counting model requires a local spatial prior to heal fragmented object boundaries, coupled with an additive signaling pathway that bypasses attention compression. Driven by this intuition, we propose ConvStack, a lightweight module that pairs localized convolutional feature extraction with deep residual fusion. By applying depthwise convolutions directly within the visual token space, ConvStack forces neighboring tokens to communicate, effectively bridging patch boundaries to construct clear, individuated object representations. To reinforce this visual evidence downstream, we inject the adapted features into the MLLM through residual connections. Zero-initialized adapter output projections preserve the input features at initialization. Our analysis gives sufficient conditions under which additive visual evidence supports stable count readout at the injection point.

Through experiments (Section 4), we evaluate how ConvStack affects counting, spatial understanding, and general visual performance. Remarkably, by fine tuning ConvStack *exclusively* on counting datasets, the model acquires substantial improvements not only in dense object counting but also across a wide range of broader spatial understanding benchmarks. The vision and language backbones remain frozen while the adapters and visual merger are trained. General visual performance remains broadly comparable, with gains on some benchmarks and decreases on others.

In summary, this paper makes three primary contributions:

- **Mechanistic Diagnosis:** Through theoretical analysis and synthetic diagnostics, we identify and formalize object individuation failure and counting aggregation collapse as two fundamental bottlenecks that systematically hinder the numeric and spatial reasoning capabilities of MLLMs.
- **Architectural Solution:** We introduce ConvStack, a lightweight convolutional module that operates directly in the visual token space to explicitly aggregate and inject local geometric structures via residual connections.
- **Empirical Validation:** We demonstrate that training ConvStack exclusively on counting tasks significantly boosts both dense counting in real images and broader spatial understanding in modern MLLMs (e.g., Qwen3-VL-8B), with broadly comparable general visual performance.

2 Related Work

Vision Transformers and convolutional bias. ViTs tokenize an image into nonoverlapping patches and rely on attention to mix spatial information (Dosovitskiy et al., 2021). This design provides global receptive fields but weakens early local inductive bias compared with CNNs (LeCun et al., 1998). Hybrid architectures reintroduce convolutional structure through convolutional attention projections, combinations of convolution and attention, and soft convolutional priors (Wu et al., 2021; Dai et al., 2021; d'Ascoli et al., 2021). Early convolutions improve ViT optimization and visual recognition (Xiao et al., 2021), while Convpass demonstrates the value of convolutional inductive bias for adaptation with few trainable parameters (Jie & Deng, 2022). ConvStack shares this emphasis on local spatial processing and convolutional adaptation. We examine how these ideas support object individuation and count readout, combining local visual features with residual routes and drawing on DeepStack's multilayer visual injection (Meng et al., 2024).

Visual counting. Counting has long been treated as a problem of localizing countable evidence and aggregating it into a cardinality estimate. Classical counting methods and methods for crowd counting often estimate density maps or evidence about individual objects before summation (Lempitsky & Zisserman, 2010), while counting everyday objects emphasizes localization of individual instances and the difficulty of natural scenes (Chattopadhyay et al., 2017). Human enumeration also distinguishes small exact counts from larger approximate number judgments (Kaufman et al., 1949; Trick & Pylyshyn, 1994; Feigenson et al., 2004), motivating examination of how counting performance varies with numerosity. We use synthetic images to isolate whether an encoder forms object units under controlled changes in patch alignment, shape, color, and size, and examine performance on small versus large counts under balanced count frequencies. These experiments probe object individuation and count readout without assuming that a model implements human cognition or exhibits a universal subitizing boundary.

Visual grounding in language models and shortcut diagnostics. Contrastive models such as CLIP (Radford et al., 2021) and MLLMs such as LLaVA (Liu et al., 2023) rely on visual encoders whose token representations are aligned to language. Spatial reasoning benchmarks and diagnostic datasets such as CLEVR (Johnson et al., 2017) show that high overall VQA accuracy can coexist with shortcut behavior and weak compositional grounding. Our controlled results make the shortcut issue concrete: a model can appear to count perfectly when the data exposes a cue from the patch grid or object area. The proposed analysis therefore treats counting accuracy as meaningful only when the construction rules out such shortcuts and requires local object individuation.

3 Method

We view visual counting as two complementary computations: *object individuation*, which forms distinct object representations, and *information aggregation*, which combines the resulting evidence to determine their number. We examine how local spatial features support object formation and how attention aggregates object evidence into a representation sensitive to count.

3.1 Patchification Makes Object Individuation Harder

ViTs learn large counts more slowly. We define visual counting as predicting $n = |\mathcal{O}(x, q)|$, where $\mathcal{O}(x, q)$ denotes the set of objects in image x specified by query q. To study counting under controlled conditions, we use SynDot and SynPoly, introduced by Che et al. (2026). These datasets contain black disks and colored polygons, respectively, on white backgrounds, with 1 to 10 objects per image in our experiments.

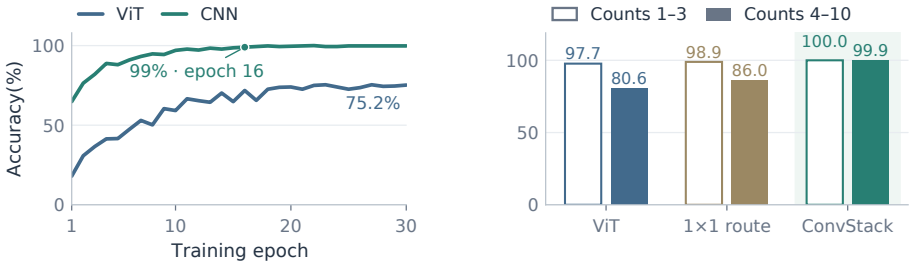


Figure 2: **Local spatial features improve counting.** Left: CNN learns faster than ViT on SynPoly. Right: ConvStack nearly closes the accuracy gap between small and large counts. The 1×1 route control leaves a substantial gap.

We begin with a simple count classification task on SynPoly to examine how the visual encoder affects counting. We compare a standard ViT and a CNN, each with four layers, training both on SynPoly with balanced count frequencies and evaluating their frozen output embeddings at each epoch using a linear classification head. Under the same training protocol, the CNN reaches 99% probe accuracy at epoch 16, whereas the ViT reaches only 75.2% by epoch 30 (Figure 2, left). The ViT achieves much lower accuracy for large counts than for small counts. In a separate three-seed comparison with balanced training counts, its mean accuracy is 97.7% for counts 1 to 3 but 80.6% for counts 4 to 10.

Moving the same disk changes the learning problem. We suspect that this difference stems from the ViT's patchification of the input image. We next use SynDot to isolate patch geometry. A ViT divides an image $x \in \mathbb{R}^{H\times W\times 3}$ into $M = HW/p^2$ nonoverlapping patches of size $p \times p$. Let P_i denote the image region of patch i and S_k the support of target object k. The fraction of that object assigned to each patch is

$$\rho_{ki} = \frac{|S_k \cap P_i|}{|S_k|}, \qquad \sum_{i=1}^{M} \rho_{ki} = 1. \tag{1}$$

We fix the disk radius to 7 pixels and patch width to 16 pixels, and vary placement relative to the grid (Figure 3). At a *patch center*, one token contains the entire disk: $\max_i \rho_{ki} = 1$. At a *shared edge*, two tokens contain equal halves: $\rho_{ki} = \rho_{kj} = 1/2$. With *random placement*, splits are generally asymmetric. The chosen sizes of disks and patches often leave a dominant fragment in one token.

The displayed center, random, and edge runs first reach 100% probe accuracy in every class at epochs 1, 9, and 14, respectively, and all finish at 100%. When each disk lies within a distinct patch, the object count equals the number of occupied patches. Splitting disks across patch boundaries breaks this direct correspondence and distributes object evidence across multiple tokens.

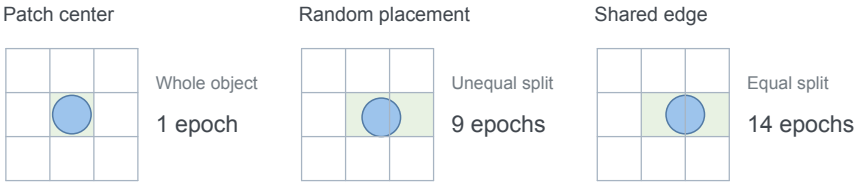


Figure 3: **Patch geometry changes learning speed.** Numbers indicate the first epoch with 100% accuracy across all count classes.

Overlapping filters provide a local prior. These observations motivate introducing a local spatial prior into the visual encoder. Standard patch embedding projects each nonoverlapping patch independently, without combining information across patch boundaries. Shared, overlapping convolutional filters aggregate neighboring pixels and provide a natural way to introduce this prior (Xiao et al., 2021; Jie & Deng, 2022). For the controlled ViT experiments, we instantiate this idea with a convolutional branch operating on image pixels that extracts hierarchical local features and injects them into the ViT through residual connections (Section B.4).

To examine the role of learned spatial filtering, we compare ConvStack with a control that replaces each 3×3 convolution with a 1×1 convolution, equivalent to a shared linear layer applied independently at each spatial location. This control retains the same depth, channel widths, pooling operations, and injection points. On SynPoly, ConvStack achieves 99.9% mean accuracy across three seeds and reduces the accuracy gap between small and large counts from 17.0 to 0.1 percentage points. The 1×1 route control retains a gap of 12.9 percentage points (Figure 2, right). These results support learned spatial filtering as a useful inductive bias for visual counting.

Table 1: **Counting on SynPoly.** Results after 15 epochs with 500 training samples per class. Metric definitions and the evaluation protocol are given in Section B.5.

Model	Accuracy	MAE	RMSE	Class sep.
ViT	0.779	0.287	0.408	1.570
CNN	0.998	0.027	0.060	2.995
ViT+ConvStack	0.998	0.009	0.037	5.835

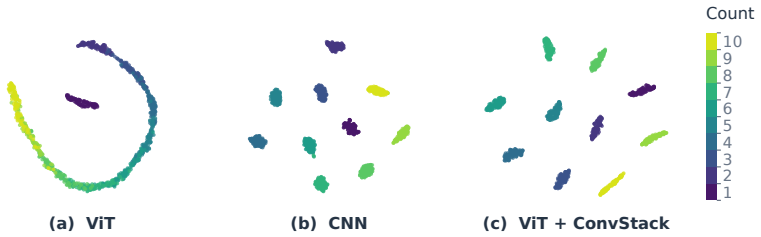


Figure 4: **ConvStack improves the separation of count classes.** t-SNE visualizations of SynPoly image embeddings, colored by object count. ConvStack forms compact, distinct clusters and reduces the class overlap seen in ViT, especially at larger counts.

ConvStack improves the structure of count classes and counting performance. We next examine how this spatial prior affects count representations by comparing ViT, CNN, and ViT+ConvStack trained on SynPoly for 15 epochs with 500 images per count class. The t-SNE (van der Maaten & Hinton, 2008) visualizations of their final image embeddings show substantial overlap among the ViT's count classes, especially at larger counts (Figure 4). Adding ConvStack produces more compact clusters of images with the same count and clearer separation between count classes. Consistent with this visual pattern, the class separation score measured on the image embeddings increases from 1.570 to 5.835, approximately 3.7× the ViT score and above the CNN's 2.995 (Table 1). Counting accuracy also rises from 77.9% to 99.8%, a gain of 21.9 percentage points, matching the CNN baseline. In the separate comparison across three seeds (Figure 2, right), ConvStack reduces the accuracy gap between small and large counts to 0.1 percentage points, versus 12.9 percentage points for the 1×1 route control. These results show that local spatial filtering strengthens visual count representations, particularly at larger counts.

3.2 Count Separation Through Visual Residuals

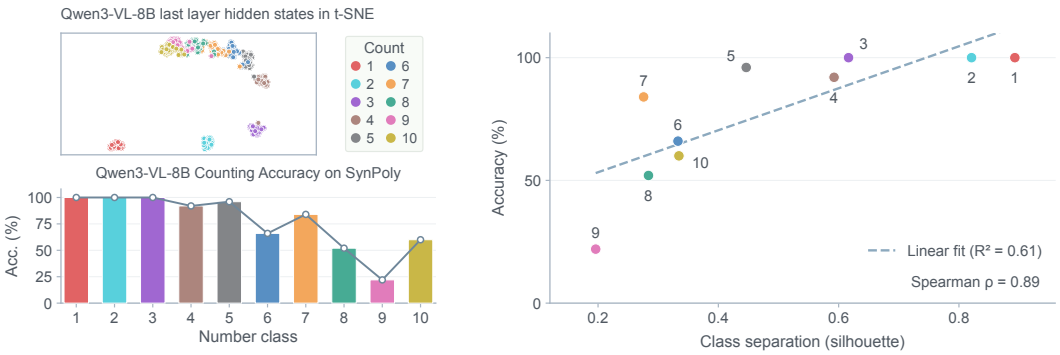


Figure 5: **Separation of count classes tracks counting accuracy.** Qwen3-VL-8B on SynPoly: t-SNE of final layer representations and accuracy for each class (left), and accuracy versus hidden state silhouette (right).

We further examine the LLM backbone of Qwen3-VL-8B and observe a positive association between mean hidden state silhouette and counting accuracy across the ten count classes ($r = 0.78$, $R^2 = 0.61$, Figure 5). We compare these representations with ConvStack using silhouette and adjacent class separation in Section D. This observation motivates us to investigate how attention aggregation (Che et al., 2026) preserves distinctions between counts.

The contribution of an additional object. To isolate aggregation from individuation, consider n objects each contained in a distinct patch: $\rho_{ki} \in \{0, 1\}$ for all $k = 1, \ldots, n$ and $i = 1, \ldots, M$, with no patch containing more than one object. Assuming one visual token per patch at the LLM input, the number of object tokens is then $N = n$. Consider a single attention head over M image tokens, with n homogeneous object tokens and fixed object and background representations (Section B.1). Writing $r = \exp(s_o - s_b)$ for the attention weight of an object token relative to a background token,

the total attention mass on object tokens is $A(n) = nr/[M + n(r-1)]$. The attention increment from one additional object is

$$\Delta A(n) = A(n+1) - A(n) = \frac{rM}{[M + n(r-1)][M + (n+1)(r-1)]}, \quad 0 \le n < M. \tag{2}$$

For attention that favors objects ($r > 1$), $\Delta A(n)$ decreases as the number of objects grows. With a fixed incoming residual h_{in} and output projection W_O, the hidden state after attention is

$$h_n = h_{\text{in}} + W_O\big[v_b + A(n)(v_o - v_b)\big]. \tag{3}$$

The difference between adjacent counts is therefore $h_{n+1} - h_n = \Delta A(n) W_O(v_o - v_b)$, giving

$$D_n = \|h_{n+1} - h_n\|_2 = K\Delta A(n), \qquad K = \|W_O(v_o - v_b)\|_2. \tag{4}$$

Thus, for $K > 0$, shrinking attention increments translate directly into smaller hidden state distances between adjacent counts, even with perfectly individuated objects. This compression helps explain why hidden state clusters for larger counts increasingly overlap and become harder to distinguish in Figure 5. Recovering exact counts from these closely spaced states requires a scalar decoder with sensitivity at least $1/D_n$ (Sections B.1 and B.2), making readout increasingly sensitive to hidden state perturbations under this model.

To provide a more reliable source of visual evidence, we adopt a residual injection strategy (Meng et al., 2024). We augment the LLM hidden state h_n after attention with a projected visual feature r_n for count n, scaled by a fixed gate g, yielding the fused state $\widetilde{h}_n$. We formalize "additive evidence" by assuming that the visual route contributes a signal along a shared unit direction that grows linearly with count, with slope $c > 2\eta \ge 0$ and error bounded by η. We also assume that the base stream's variation between adjacent counts along this direction is bounded by β. Under these conditions, the fused state $\widetilde{h}_n$ and its effective separation margin κ become:

$$\widetilde{h}_n = h_n + gr_n, \qquad \kappa = |g|(c - 2\eta) - \beta, \qquad \text{for } g \neq 0. \tag{5}$$

Theorem 1 (Stable count readout from preserved residual evidence)**.** *Assuming the evidence and interference bounds outlined above hold uniformly up to some maximum count $n_{\max}$ (Assumption 2), any configuration where $\kappa > 0$ ensures the residual states are separated by a margin completely independent of the count itself. Under these conditions, there exists an exact scalar decoder $F : \mathbb{R}^d \to \mathbb{R}$ with $F(\widetilde{h}_n) = n$ such that:*

$$\|\widetilde{h}_m - \widetilde{h}_n\|_2 \ge \kappa|m - n|, \qquad \text{Lip}(F) \le 1/\kappa. \tag{6}$$

By rounding F to the nearest integer, one can exactly recover the true count even when the states are corrupted by any additive noise with ℓ_2 norm strictly less than $\kappa/2$.

When the additive visual signal outweighs the base stream's interference ($\kappa > 0$), the theorem guarantees a count separation margin and the existence of a decoder with sensitivity at most $1/\kappa$ throughout the stated count range.

This is a sufficient condition at the injection point. Real images need not satisfy the evidence assumptions, and the theorem does not guarantee that training learns these features or that subsequent language layers preserve them (Section B.3). It motivates extracting local visual cues and reinforcing them at multiple network depths.

3.3 The ConvStack Architecture

Driven by the need to continuously inject stable counting evidence, we propose ConvStack, an architecture that pairs localized convolutional feature extraction with a deep residual fusion mechanism. To enhance the spatial reasoning capabilities of pretrained MLLMs, ConvStack operates directly in the visual token space (Figure 6).

At each adapter location, we reshape the visual tokens into a 2D grid X and update it with a convolutional bottleneck:

$$\mathcal{C}_\theta(X) = W_{\text{up}} * \text{GELU}\left(K_{\text{dw}} * \text{GELU}(W_{\text{down}} * \text{LN}(X))\right), \qquad \widetilde{X} = X + \alpha\mathcal{C}_\theta(X). \tag{7}$$

Here, W_{down} and W_{up} are 1×1 projections that compress and restore the channel dimension, while the 3×3 depthwise convolution K_{dw} mixes neighboring tokens in the compressed space. A learnable scalar α scales the correction. The updated features $\widetilde{X}$ are flattened into visual tokens for fusion with the hidden states at selected LLM layers. We initialize W_{up} to zero so that each adapter starts as an identity mapping, preserving the pretrained features at initialization.

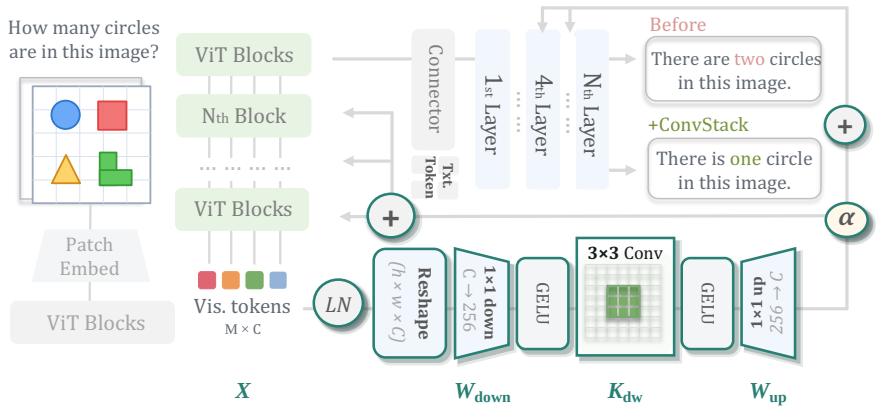


Figure 6: **Overview of ConvStack.** Teal outlines highlight the ConvStack components. Variable names correspond to Equation (7).

4 Experiments

4.1 Experimental Setup

Data and baselines. We implement ConvStack on Qwen3-VL-8B (Bai et al., 2025) and compare it with Spatial-MLLM-3B (Wu et al., 2025), LLaVA-SP Cropping-7B (Lou et al., 2025), and Honeybee-7B (Cha et al., 2024). We assess counting in real images on PixMo-Count (Deitke et al., 2025) and CountBenchQA (Beyer et al., 2024). We evaluate spatial understanding on CV-Bench (Tong et al., 2024), SAT-real (Ray et al., 2024), SpatialEval (Wang et al., 2024), and SAT-Spatial (hunarbatra, 2025). We assess general visual capability on POPE (Li et al., 2023), MME (Fu et al., 2026), MMBench-EN (Liu et al., 2024), MMMU (Yue et al., 2024), RealWorldQA (xAI, 2024), and MathVista (Lu et al., 2024). Evaluation uses deterministic decoding and each model's image processor. Dataset descriptions, evaluation splits, and training variants are detailed in Section A.

Training. Our standard training mixture combines examples from the PixMo-Count training split with 2,000 SynPoly and 2,000 SynDot examples, following the synthetic datasets introduced by Che et al. (2026). For Qwen3-VL, we train convolutional adapters at vision layer 18 (ablation in Section C.1) and all native DeepStack levels, together with the visual merger. The remaining pretrained parameters are frozen. We train for one epoch with an effective batch size of 32, a learning rate of 5×10^{-5}, and loss on tokens in counting answers.

4.2 Evaluation

Counting and Spatial Understanding. As shown in Table 2, ConvStack-8B consistently outperforms both the base Qwen3-VL-8B and alternative spatial adapters. The most striking gains are in dense counting, where it achieves 75.19% on PixMo-Count, an absolute improvement of 9.73 percentage points over the base model (65.46%). Furthermore, the injected local features systematically benefit broader spatial reasoning, establishing top scores on benchmarks like CV-Bench Spatial (93.20%) and SAT-Spatial (76.81%). Notably, ConvStack proves superior to existing workarounds like Pool SFE or Crop+DFI (Lou et al., 2025), and avoids the severe performance collapse caused by aggressive strategies for token reduction, such as C-Abstractor (Cha et al., 2024).

General Visual Capability. As shown in Table 3, ConvStack-8B maintains broadly comparable general visual performance to Qwen3-VL-8B, with improvements on four of the six benchmarks. POPE F1 rises from 87.83% to 89.52%, MME from 2219.6 to 2251.0, and RealWorldQA from 69.93% to 70.33%, while MMBench and MathVista show small decreases.

Table 2: **Evaluation on counting and spatial understanding benchmarks.** CV-Bench Spatial averages Relation, Depth, and Distance accuracy. The best result is **bolded**, and the second best is underlined. Dashes denote results that are unavailable or incompatible with the scorer.

Method	PixMo-Count	CountBenchQA	CV-Bench Spatial	SAT-real	SpatialEval	SAT-Spatial
Models Specialized for Spatial Tasks						
Spatial-MLLM-3B	52.67	65.78	75.22	60.00	50.00	58.76
LLaVA-SP Cropping-7B	42.94	45.62	63.59	52.00	31.33	60.61
Honeybee-7B (C-Abstractor)	37.98	54.79	60.50	51.00	28.00	57.63
Qwen3-VL-8B Scale						
Qwen3-VL-8B	65.46	89.82	92.34	59.33	61.33	76.79
Qwen3-VL-8B + Crop+DFI	71.43	85.74	93.04	62.13	65.22	–
Qwen3-VL-8B + Pool SFE	73.52	89.21	92.40	58.67	64.64	75.05
Qwen3-VL-8B + C-Abstractor	36.95	29.53	–	29.33	30.46	–
ConvStack-8B	**75.19**	**90.22**	**93.20**	**63.33**	**66.67**	**76.81**

Table 3: **General visual capability.** POPE reports F1. Scores are percentages except MME. The better result in each column is bolded.

Method	POPE	MME	MMBench	MMMU	RealWorldQA	MathVista
Qwen3-VL-8B	87.83	2219.6	**88.67**	59.00	69.93	**66.60**
ConvStack-8B	**89.52**	**2251.0**	88.60	**59.89**	**70.33**	65.00

4.3 Ablation Study

Table 4 compares ConvStack against three alternative fine tuning strategies on Qwen3-VL-8B. All variants share the exact same training data and hyperparameters. *ViT-stack only* trains the ConvStack with ViT layer residual connections only. *LLM-stack only* directly routes the ConvStack head output to the residual connection of the LLM backbone. 1×1 *convolution kernel* replaces the 3×3 depthwise convolution with a 1×1 projection, removing spatial mixing within the adapter. *LoRA* applies LoRA throughout the 8B MLLM. Our method trains 42.8M parameters in its residual adapters and visual merger. The results confirm that explicit local token mixing (3×3 convolution) is critical.

Table 4: Ablation study (accuracy %).

Configuration	PixMo-Count	SAT-real
ConvStack-8B	**75.19**	**63.33**
ViT-stack only	73.09	60.67
LLM-stack only	74.05	61.33
1×1 convolution kernel	73.09	59.33
LoRA	73.85	62.00

4.4 Mechanistic Analysis

Internal representations of counting. To understand how ConvStack improves counting, we analyze its internal representations using 500 SynPoly images with object counts ranging from 1 to 10. As shown in Figure 7, ConvStack produces clearer clusters for each count, alongside an accuracy increase from 56.8% to 98.8% for counts 6 to 10. In the original hidden space, mean cosine silhouette rises from 0.479 to 0.707, and separation improves for every adjacent count pair (Section D).

Probing spatial features via S-Space. To verify that these representation benefits extend beyond counting to general spatial grounding, we apply S-Space (MirroS Team, 2026) to probe intermediate model representations on 500 MSCOCO images. S-Space allows us to linearly read out object coordinates (horizontal, vertical, and depth) directly from the hidden states.

Figure 8(a) highlights a representative failure of the base model, which incorrectly judges the dining table to be farther away than the refrigerator. ConvStack corrects this error, and looking at the

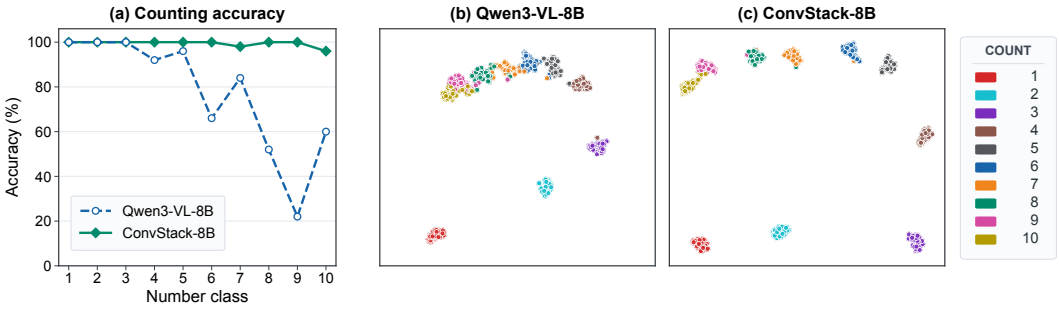


Figure 7: **Counting accuracy and representations on SynPoly.** (a) Accuracy by number class, with 50 images per class. (b,c) t-SNE of the final layer representation of the last prompt token for counts 1 to 10.

internal S-Space readout (Figure 8(b)), we see the corresponding depth probe shift dramatically from -0.28 (incorrectly farther) to $+0.08$ (correctly closer). Zooming out to the full set of 500 images (Figure 8(c)), ConvStack yields consistent aggregate gains in representational accuracy across all three dimensions ($+3.5\%$ horizontal, $+2.3\%$ vertical, and $+2.1\%$ depth). This confirms that explicitly mixing local visual features forces the network to maintain a much stronger internal grounding of physical object relations.

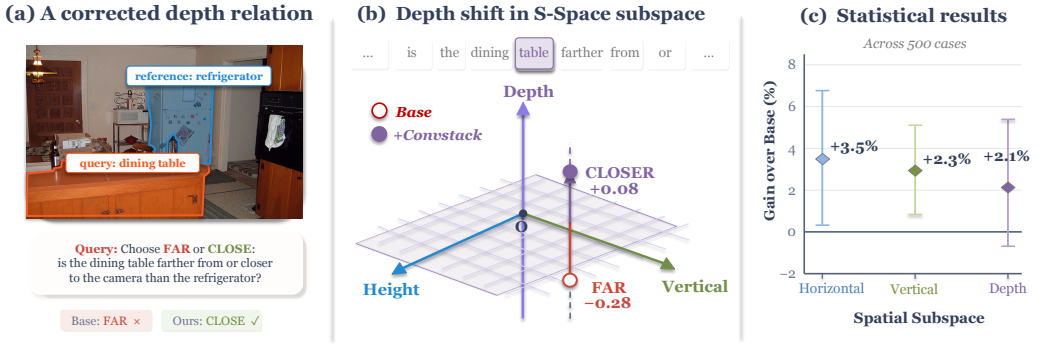


Figure 8: **Spatial understanding analyzed with S-Space on** 500 **MSCOCO images.** (a) ConvStack corrects the base model's depth judgment for a dining table relative to a refrigerator. (b) The corresponding S-Space depth readout shifts from -0.28 (farther) to $+0.08$ (closer). (c) Aggregate gains over the base model are positive for horizontal, vertical, and depth evaluations.

5 Conclusion

In this work, we identified two critical bottlenecks in the object individuation and counting aggregation processes that cause MLLMs to struggle with visual counting, especially for large numbers. To address these limitations, we proposed ConvStack, an architecture that effectively improves the separation of internal counting representations. Remarkably, by training exclusively on counting tasks, the model achieves significant improvements in both visual counting and broader spatial understanding. These gains are accompanied by broadly comparable general visual performance, supporting ConvStack as a practical approach to strengthening numeric and spatial reasoning in multimodal foundation models.

AI USE STATEMENT

In this work, we used generative AI tools to implement methods, assist with translation and writing, and clean and reformat datasets. We have reviewed all work assisted by AI. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report, 2025. URL https://arxiv.org/abs/2511.21631.

Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, et al. PaliGemma: A versatile 3B VLM for transfer. *arXiv preprint arXiv:2407.07726*, 2024.

Junbum Cha, Wooyoung Kang, Jonghwan Mun, and Byungseok Roh. Honeybee: Locality-enhanced projector for multimodal LLM. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Prithvijit Chattopadhyay, Ramakrishna Vedantam, Ramprasaath R. Selvaraju, Dhruv Batra, and Devi Parikh. Counting everyday objects in everyday scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1135–1144, 2017.

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, 2026. URL https://arxiv.org/abs/2603.18523.

Zihang Dai, Hanxiao Liu, Quoc V. Le, and Mingxing Tan. Coatnet: Marrying convolution and attention for all data sizes. In *Advances in Neural Information Processing Systems*, 2021.

Stéphane d'Ascoli, Hugo Touvron, Matthew L. Leavitt, Ari S. Morcos, Giulio Biroli, and Levent Sagun. Convit: Improving vision transformers with soft convolutional inductive biases. In *International Conference on Machine Learning*, 2021.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, Jiasen Lu, Taira Anderson, Erin Bransom, Kiana Ehsani, Huong Ngo, YenSung Chen, Ajay Patel, Mark Yatskar, Chris Callison-Burch, Andrew Head, Rose Hendrix, Favyen Bastani, Eli VanderBilt, Nathan Lambert, Yvonne Chou, Arnavi Chheda, Jenna Sparks, Sam Skjonsberg, Michael Schmitz, Aaron Sarnat, Byron Bischoff, Pete Walsh, Chris Newell, Piper Wolters, Tanmay Gupta, Kuo-Hao Zeng, Jon Borchardt, Dirk Groeneveld, Crystal Nam, Sophie Lebrecht, Caitlin Wittlif, Carissa Schoenick, Oscar Michel, Ranjay Krishna, Luca Weihs, Noah A. Smith, Hannaneh Hajishirzi, Ross Girshick, Ali Farhadi, and Aniruddha Kembhavi. Molmo and PixMo: Open Weights and Open Data for State-of-the-Art Vision-Language Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 91–104, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Deitke_Molmo_and_PixMo_Open_Weights_and_Open_Data_for_State-of-the-Art_CVPR_2025_paper.html.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

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

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *Advances in Neural Information Processing Systems*, 38, 2026.

Google DeepMind. Gemma — google deepmind. https://deepmind.google/models/gemma/, 2024. Accessed: 2026-09-18.

Xuyang Guo, Zekai Huang, Zhenmei Shi, Zhao Song, and Jiahao Zhang. Your vision-language model can't even count to 20: Exposing the failures of vlms in compositional counting. *arXiv preprint arXiv:2510.04401*, 2025.

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. *arXiv preprint arXiv:2511.17699*, 2025.

hunarbatra. SAT-Spatial-VQA. Hugging Face dataset, 2025. URL https://huggingface.co/datasets/hunarbatra/SAT-Spatial-VQA.

Shibo Jie and Zhi-Hong Deng. Convolutional bypasses are better vision transformer adapters. *arXiv preprint arXiv:2207.07039*, 2022.

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 *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2901–2910, 2017.

E. L. Kaufman, M. W. Lord, T. W. Reese, and J. Volkmann. The discrimination of visual number. *The American Journal of Psychology*, 62(4):498–525, 1949.

Yann LeCun, Leon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Victor Lempitsky and Andrew Zisserman. Learning to count objects in images. In *Advances in Neural Information Processing Systems*, 2010.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 292–305, 2023.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916, 2023.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pp. 216–233. Springer, 2024.

Haoran Lou, Chunxiao Fan, Ziyan Liu, Yuexin Wu, and Xinliang Wang. LLaVA-SP: Enhancing visual representation with visual spatial tokens for MLLMs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22014–22024, October 2025.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=KUNzEQMWU7.

Lingchen Meng, Jianwei Yang, Rui Tian, Xiyang Dai, Zuxuan Wu, Jianfeng Gao, and Yu-Gang Jiang. DeepStack: Deeply stacking visual tokens is surprisingly simple and effective for LMMs. *Advances in Neural Information Processing Systems*, 37:23464–23487, 2024.

MirroS Team. S-Space: Exploring spatial workspace in multimodal models, September 2026. URL https://mirros.ai/blog/s-space. Blog post.

Roni Paiss, Ariel Ephrat, Omer Tov, Shiran Zada, Inbar Mosseri, Michal Irani, and Tali Dekel. Teaching CLIP to Count to Ten. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3170–3180, 2023. URL https://openaccess.thecvf.com/content/ICCV2023/html/Paiss_Teaching_CLIP_to_Count_to_Ten_ICCV_2023_paper.html.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763, 2021.

Arijit Ray, Jiafei Duan, Ellis Brown, Reuben Tan, Dina Bashkirova, Rose Hendrix, Kiana Ehsani, Aniruddha Kembhavi, Bryan A Plummer, Ranjay Krishna, et al. Sat: Dynamic spatial aptitude training for multimodal language models. *arXiv preprint arXiv:2412.07755*, 2024.

Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai C Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *Advances in Neural Information Processing Systems*, 37: 87310–87356, 2024.

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):80–102, 1994.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL https://www.jmlr.org/papers/v9/vandermaaten08a.html.

Jiayu Wang, Yifei Ming, Zhenmei Shi, Vibhav Vineet, Xin Wang, Yixuan Li, and Neel Joshi. Is a picture worth a thousand words? delving into spatial reasoning for vision language models. *Advances in Neural Information Processing Systems*, 37:75392–75421, 2024.

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.

Diankun Wu, Fangfu Liu, Yi-Hsin Hung, and Yueqi Duan. Spatial-MLLM: Boosting MLLM capabilities in visual-based spatial intelligence. *arXiv preprint arXiv:2505.23747*, 2025.

Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. In *IEEE/CVF International Conference on Computer Vision*, pp. 22–31, 2021.

xAI. RealWorldQA. Hugging Face dataset, 2024. URL https://huggingface.co/datasets/xai-org/RealworldQA. Accessed: 2026-09-25.

Tete Xiao, Mannat Singh, Eric Mintun, Trevor Darrell, Piotr Dollár, and Ross Girshick. Early convolutions help transformers see better. In *Advances in Neural Information Processing Systems*, 2021.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. MMMU: A Massive Multi-discipline Multimodal Understanding and Reasoning Benchmark for Expert AGI. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9556–9567, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Yue_MMMU_A_Massive_Multi-discipline_Multimodal_Understanding_and_Reasoning_Benchmark_for_CVPR_2024_paper.html.

APPENDIX

A DATASETS

We describe the datasets used for training, counting diagnostics, and evaluation. The standard adaptation mixture uses the PixMo-Count training split and 2,000 examples each from SynPoly and SynDot.

A.1 COUNTING DATA AND CONTROLLED DIAGNOSTICS

PixMo-Count. PixMo-Count (Deitke et al., 2025) contains natural images annotated with a target object category and its count, with point annotations provided for training. Its validation and test splits, verified by human annotators, cover counts from 2 to 10. We use examples from the training split for counting adaptation and evaluate accuracy for exact counts on the available test images. The current Qwen3 evaluation contains 524 test examples.

SynPoly. SynPoly, introduced by Che et al. (2026), contains colored polygons on a white background with variation in shape, size, color, and position. It tests counting under appearance variation while keeping the scene simple. We use 2,000 examples in the standard adaptation mixture. The controlled encoder experiments use balanced counts from 1 to 10 and resolutions chosen for each experiment. The counting suite for pretrained models additionally uses 380 generated test images at 512×512 resolution, with 20 examples for each count from 1 to 19. The representation analysis in Figure 7 uses a separate set of 500 images, with 50 examples per count from 1 to 10.

SynDot. SynDot, also introduced by Che et al. (2026), contains black disks on a white background. Its simple appearance makes it useful for isolating the relationship between object placement and patch geometry. We use 2,000 examples in the standard adaptation mixture. For controlled diagnostics, we construct variants with objects at patch centers, at patch edges, or at random positions, with counts from 1 to 10. The counting suite for pretrained models uses 380 generated 512×512 test images covering counts from 1 to 19, with 20 examples per count.

CountBenchQA. CountBenchQA (Beyer et al., 2024) converts CountBench (Paiss et al., 2023) into a visual question answering task by adding questions about object quantity. The original images contain between 2 and 10 instances of a target object. Our additional counting evaluations use the public test release of 491 images and accuracy for exact counts.

A.2 SPATIAL UNDERSTANDING BENCHMARKS

CV-Bench. CV-Bench (Tong et al., 2024) recasts annotated images from ADE20K, COCO, and Omni3D as visual questions about counting, spatial relations, depth order, and relative distance. Our CV-Bench Spatial score averages accuracy equally over Relation (650 questions), Depth (600), and Distance (600), covering 1,850 questions. The counting category is excluded from this spatial score.

SAT-real. SAT-real is the test split of Spatial Aptitude Training (SAT) that uses real images (Ray et al., 2024) evaluating spatial relationships and changes involving objects or viewpoints, with one or two images per question.

SpatialEval. SpatialEval (Wang et al., 2024) evaluates spatial relationships, position, counting, and navigation through tasks involving maps, grids, mazes, and real images. Our experiments use a fixed subset of 150 visual questions with multiple choices shared across models and report answer accuracy on this subset.

SAT-Spatial. We use the public SAT-Spatial-VQA release (hunarbatra, 2025), which contains 15,000 examples for spatial reasoning, each consisting of an image, a question, and an answer. Although the release names its split `train`, we use these examples only for evaluation. We retain the native questions and score normalized exact matches to the answer values.

A.3 General Visual Benchmarks

POPE. POPE (Li et al., 2023) measures object hallucination through yes/no questions about whether a specified object appears in an image. We use the random, popular, and adversarial subsets based on COCO, each containing 3,000 questions, and report F1 over all 9,000 questions.

MME. MME (Fu et al., 2026) evaluates perception and cognition across 14 tasks, including object recognition, spatial position, text recognition, and reasoning. Our evaluation contains 2,374 yes/no questions arranged in 1,187 pairs. Each task combines accuracy on individual questions with the accuracy of answering both questions in a pair correctly. We report the sum of perception and cognition scores.

MMBench-EN. MMBench (Liu et al., 2024) is a benchmark with multiple choices per question covering a broad range of visual perception and reasoning abilities. We use the cached English development set with 4,377 evaluation records and report accuracy using deterministic option extraction.

MMMU. MMMU (Yue et al., 2024) tests multimodal understanding and reasoning at an expert level across 30 academic subjects. We evaluate the validation split of 900 questions, which includes both questions with multiple choices and questions requiring free responses, and report accuracy with answer normalization for free responses.

RealWorldQA. RealWorldQA (xAI, 2024) evaluates understanding of everyday visual scenes, including photographs taken from vehicles. We use its test split of 765 examples, containing both questions with multiple choices and questions requiring short answers, and report accuracy after option extraction or answer normalization.

MathVista. MathVista (Lu et al., 2024) evaluates mathematical reasoning grounded in visual inputs such as diagrams, charts, and natural images. We use the `testmini` split of 1,000 examples and report accuracy with local answer extraction followed by normalization of answer types and numerical precision.

A.4 Images for Spatial Representation Analysis

MSCOCO. MSCOCO (Lin et al., 2014) provides natural images of everyday scenes with annotations of object instances. We select 500 images for the S-Space analysis (MirroS Team, 2026) in Section 4.4, comparing horizontal, vertical, and depth information in the base model and its ConvStack adaptation. These images form a set for representation analysis.

B Method and Analysis

B.1 Attention Model and Compression of Count Margins

Assumption 1 (Homogeneous object and background tokens)**.** The head attends to a fixed number $M \geq 2$ of visual tokens. The LLM receives perfectly individuated objects: each of the $n \in \{0, \dots, M\}$ objects occupies a distinct image token, so the number of object tokens is $N = n$. The n object tokens share logit s_o and value v_o. The remaining $M - n$ tokens share logit s_b and value v_b. These logits and values, the output projection W_O, and the incoming residual h_{in} are independent of n. Write $r = \exp(s_o - s_b)$.

Proposition 1 (Compression of count margins under selective attention)**.** *Under Assumption 1, the total attention mass on object tokens is $A(n) = nr/[M + n(r-1)]$, with the increment between adjacent counts given by Equation (2). For a head that favors objects ($r > 1$), $A(n)$ is strictly increasing and discretely strictly concave: $\Delta A(n+1) < \Delta A(n)$ for $0 \leq n < M - 1$. Uniform attention ($r = 1$) instead gives the constant interval $\Delta A(n) = 1/M$.*

Proof. Under Assumption 1, the softmax denominator is $ne^{s_o} + (M-n)e^{s_b}$. Summing the weights of the n object tokens and dividing by e^{s_b} gives $A(n) = nr/[M + n(r-1)]$. Let $d_n = M + n(r-1)$, which is positive for $0 \leq n \leq M$ and $r > 0$. Direct subtraction yields

$$A(n+1) - A(n) = r\frac{(n+1)d_n - nd_{n+1}}{d_n d_{n+1}} = \frac{rM}{d_n d_{n+1}} > 0. \tag{B.1}$$

For $r > 1$ and $0 \le n < M - 1$,

$$\Delta A(n+1) - \Delta A(n) = -\frac{2rM(r-1)}{d_n d_{n+1} d_{n+2}} < 0. \tag{B.2}$$

This proves strict discrete concavity. If $r = 1$, then $d_n = M$ and $\Delta A(n) = 1/M$. □

The inverse quadratic approximation follows from the exact factorization, for $n \ge 1$ and $r > 1$,

$$\Delta A(n) = \frac{rM}{(r-1)^2 n(n+1)} \left(1 + \frac{M}{n(r-1)}\right)^{-1} \left(1 + \frac{M}{(n+1)(r-1)}\right)^{-1}. \tag{B.3}$$

Both correction factors are close to one when $n(r-1) \gg M$. This is an approximation within the finite count range $n < M$, rather than a limit that sends n beyond the token budget.

B.2 Decoder Sensitivity Under Compressed Count Intervals

Under Assumption 1, $N = n$ and the hidden state is given by Equation (3). Its distance between adjacent counts satisfies $D_n = K\Delta A(n)$ as stated in Equation (4), where $K = \|W_O(v_o - v_b)\|_2$. If $D_n > 0$, every scalar decoder exact on h_n, h_{n+1} has Lipschitz constant at least $1/D_n$. A decoder can distinguish these two counts under every additive perturbation of ℓ_2 norm at most ε if and only if $D_n > 2\varepsilon$. If $D_n \le 2\varepsilon$, every scalar decoder has absolute error at least $1/2$ for one of the two counts at some admissible perturbed state. When $K = 0$, the two unperturbed states coincide.

Proof. The total object and background weights are $A(n)$ and $1 - A(n)$, so $o_n = v_b + A(n)(v_o - v_b)$. The fixed incoming residual cancels between adjacent states:

$$h_{n+1} - h_n = \Delta A(n) W_O(v_o - v_b). \tag{B.4}$$

Taking Euclidean norms proves Equation (4). For any exact scalar decoder with finite Lipschitz constant L_F,

$$1 = |F(h_{n+1}) - F(h_n)| \le L_F \|h_{n+1} - h_n\|_2 = L_F D_n. \tag{B.5}$$

Hence $L_F \ge 1/D_n$ whenever $D_n > 0$. A decoder without a finite Lipschitz constant satisfies the bound trivially.

Define the full closed perturbation balls

$$B_n(\varepsilon) = \{z : \|z - h_n\|_2 \le \varepsilon\}. \tag{B.6}$$

If $D_n > 2\varepsilon$, the balls around h_n and h_{n+1} are disjoint. The decoder that selects the nearest center distinguishes them: for $z \in B_n(\varepsilon)$,

$$\|z - h_{n+1}\|_2 \ge D_n - \|z - h_n\|_2 > \varepsilon \ge \|z - h_n\|_2, \tag{B.7}$$

and symmetrically for $z \in B_{n+1}(\varepsilon)$.

Conversely, if $D_n \le 2\varepsilon$, the midpoint $z_\star = (h_n + h_{n+1})/2$ lies in both balls. A single decoder output cannot equal both labels there. For every scalar decoder F, the triangle inequality gives

$$1 \le |F(z_\star) - n| + |F(z_\star) - (n+1)|, \tag{B.8}$$

so at least one of the two absolute errors is at least $1/2$. Finally, $K = 0$ implies $h_{n+1} = h_n$, making exact decoding impossible even at $\varepsilon = 0$. □

The result is pairwise and conditional on Assumption 1. For a finite family of count states, a decoder that selects the nearest center is robust on all full perturbation balls if their minimum pairwise distance exceeds 2ε. In the homogeneous model with $K > 0$, the states lie in order on a line, so the minimum pairwise distance is the smallest adjacent distance. Heterogeneous values, additional attention heads, or residuals that depend on count can change this geometry and preserve other count cues. Furthermore, actual image perturbations need not fill a norm ball: ball overlap proves an obstruction in the worst case for that uncertainty model, not the existence of a particular image transformation.

B.3 Proof of Theorem 1

Assumption 2 (Approximately additive residual evidence and bounded interference)**.** For a finite family of count states $n = 0, \dots, n_{\max}$, $n_{\max} \geq 1$, let $h_n, r_n \in \mathbb{R}^d$ be the LLM hidden state before visual residual injection and the projected feature from the visual route, with fixed gate $g \neq 0$. There exist a unit vector u and constants $a \in \mathbb{R}$, $c > 2\eta \geq 0$, $\beta \geq 0$, shared across all counts, such that

$$u^\top r_n = cn + a + \xi_n, \qquad |\xi_n| \leq \eta, \tag{B.9}$$

$$|u^\top (h_{n+1} - h_n)| \leq \beta, \qquad 0 \leq n < n_{\max}. \tag{B.10}$$

With $\widetilde{h}_n$ and κ defined in Equation (5), these assumptions imply Theorem 1 whenever $\kappa > 0$. The constructed decoder also satisfies, for every $\|\delta\|_2 \leq \varepsilon$,

$$|F(\widetilde{h}_n + \delta) - n| \leq \varepsilon/\kappa. \tag{B.11}$$

Proof. Let $s = \operatorname{sign}(g)$ and project the residual states onto the unit direction su: $t_n = su^\top \widetilde{h}_n$. Using Equations (B.9) and (B.10),

$$\begin{aligned} t_{n+1} - t_n &= su^\top (h_{n+1} - h_n) + |g|\big(c + \xi_{n+1} - \xi_n\big) \\ &\geq -\beta + |g|(c - 2\eta) = \kappa > 0. \end{aligned} \tag{B.12}$$

Thus $t_0 < \cdots < t_{n_{\max}}$. For $m > n$, telescoping the increments yields

$$\|\widetilde{h}_m - \widetilde{h}_n\|_2 \geq |u^\top (\widetilde{h}_m - \widetilde{h}_n)| = t_m - t_n \geq \kappa(m - n). \tag{B.13}$$

Symmetry gives Equation (6) for all pairs.

Define a continuous scalar function on the real line by

$$f(t) = \begin{cases} 0, & t \leq t_0, \\ n + \dfrac{t - t_n}{t_{n+1} - t_n}, & t_n \leq t \leq t_{n+1}, \quad 0 \leq n < n_{\max}, \\ n_{\max}, & t \geq t_{n_{\max}}. \end{cases} \tag{B.14}$$

Each linear segment has slope between 0 and $1/\kappa$, and the expressions agree at their endpoints. Hence f has global Lipschitz constant at most $1/\kappa$. The decoder $F(x) = f(su^\top x)$ satisfies $F(\widetilde{h}_n) = n$ and

$$|F(x) - F(y)| \leq \frac{|u^\top (x - y)|}{\kappa} \leq \frac{\|x - y\|_2}{\kappa}. \tag{B.15}$$

Applying this inequality with $x = \widetilde{h}_n + \delta$ and $y = \widetilde{h}_n$ proves Equation (B.11). For $\varepsilon < \kappa/2$, the error is strictly less than $1/2$, so rounding to the nearest integer returns n. □

The theorem applies directly to residual addition in the original state space. Its preservation condition bounds interference only along the shared evidence direction. Orthogonal changes in the base stream need not be small. The same direction and constants must work across the entire stated count family. For a collection with multiple images per count, coverage requires those images' representations to lie in the corresponding perturbation neighborhoods, or a separate uniform separation argument. Neither this coverage nor the required approximate additivity is guaranteed by convolution alone. The theorem establishes the existence of a stable readout at the injection representation, not that optimization finds it or that all later transformations preserve it.

A visual route can use information that is weak in the aggregated state. In contrast, a residual depending only on that state, $\widehat{h} = h + gC(h)$ with C having Lipschitz constant L_C, obeys

$$\|\widehat{h}_{n+1} - \widehat{h}_n\|_2 \leq (1 + |g|L_C)\|h_{n+1} - h_n\|_2. \tag{B.16}$$

Thus a residual with bounded sensitivity that acts only on an already compressed state has bounded distance amplification and cannot separate identical states. Access to visual features retaining complementary count evidence is therefore relevant to the proposed route. Its usefulness remains an empirical question for the learned adapters.

B.4 Object Formation and Aggregation Error

Let $\widehat{\mathcal{O}}(x, q)$ be a conceptual set of predicted object units, $\hat{m} = |\widehat{\mathcal{O}}(x, q)|$, and $n = |\mathcal{O}(x, q)|$. For a predicted count $\hat{n}$, the triangle inequality gives the valid decomposition

$$\underbrace{|\hat{n} - n|}_{\mathcal{E}_{\text{count}}} \leq \underbrace{|\hat{m} - n|}_{\mathcal{E}_{\text{unit}}} + \underbrace{|\hat{n} - \hat{m}|}_{\mathcal{E}_{\text{agg}}}. \tag{B.17}$$

Equality is not guaranteed because errors in the two stages can cancel. Here $\mathcal{E}_{\text{unit}}$ measures only the cardinality error of the formed units. It does not detect incorrect object identities or compensating splits and merges. The decomposition separates two possible failure sources without asserting that one universally dominates the other.

For the patch fractions in Equation (1), define the fragmentation index

$$J_k = 1 - \sum_{i=1}^{M} \rho_{ki}^2. \tag{B.18}$$

Because the fractions are nonnegative and sum to one, $0 \leq J_k \leq 1 - 1/M$. A disk contained in one patch has $J_k = 0$. A disk split equally between two patches has $J_k = 1/2$. An unequal split between two patches has $J_k = 2\rho(1 - \rho) < 1/2$. This statistic describes how object support is distributed across tokens, without asserting a convergence rate or an ordering for arbitrary placements across multiple patches.

Patch occupancy is sufficient for counting only when each object occupies one patch and each occupied patch contains one object. Even then, linear decodability requires a feature condition. For example, suppose the initial embeddings after correcting for position $\bar{z}_i = z_i^{(0)} - e_i$ obey a shared margin condition: there exist u, t, and $\gamma > 0$ such that

$$u^\top \bar{z}_i - t \geq \gamma \quad \text{on occupied patches,} \qquad u^\top \bar{z}_i - t \leq -\gamma \quad \text{on background patches.} \tag{B.19}$$

Then $\sum_i \mathbf{1}\{u^\top \bar{z}_i > t\} = n$. Patch alignment alone does not prove that an arbitrary learned projection satisfies this condition.

Objects with diverse appearances or objects crossing patch boundaries require grouping several fragments while separating nearby instances. The pixel hierarchy used in the controlled ViT experiments has four stages, each consisting of a 3×3 convolution, ReLU, and 2×2 max pooling. Each stage's features are projected with a 1×1 convolution, adaptively pooled to the ViT patch grid, normalized, and added to the patch tokens before the corresponding transformer block through a scalar gate. The pretrained adapters follow Equation (7). Their injection points and training setup are specified in Section 4.1. These architectural priors motivate the method. Neither exact additive evidence nor a strict improvement in information content is assumed.

B.5 Metrics for Controlled Counting Experiments

For Table 1, we evaluate each model on $T = 1{,}000$ SynPoly images, with 100 images for each count from 1 to 10. Let $y_i \in \{1, \ldots, 10\}$ be the true count of image i and $p_i(k)$ its predicted probability for count k. The controlled models obtain these probabilities by applying softmax to the scaled similarities between the image embedding and ten text prototypes, one for each count. Each prototype is the normalized average of 20 caption embeddings for that count.

Accuracy. We use the most probable count class as the discrete prediction:

$$\hat{y}_i^{\text{argmax}} = \operatorname*{arg\,max}_{k \in \{1, \ldots, 10\}} p_i(k), \qquad \text{Accuracy} = \frac{1}{T} \sum_{i=1}^{T} \mathbf{1}\{\hat{y}_i^{\text{argmax}} = y_i\}. \tag{B.20}$$

Accuracy is reported as a fraction in Table 1.

MAE and RMSE. Both error metrics use the continuous expected count from the same predictive distribution,

$$\hat{y}_i^{\text{exp}} = \sum_{k=1}^{10} k\, p_i(k), \tag{B.21}$$

without rounding:

$$\mathrm{MAE} = \frac{1}{T}\sum_{i=1}^{T}|\hat{y}_i^{\mathrm{exp}} - y_i|, \qquad \mathrm{RMSE} = \sqrt{\frac{1}{T}\sum_{i=1}^{T}(\hat{y}_i^{\mathrm{exp}} - y_i)^2}. \tag{B.22}$$

Thus, accuracy measures exact classification, while MAE and RMSE measure the deviation of the expected count.

Class separation. Let $z_i \in \mathbb{R}^{128}$ be the final image embedding, and let $\mathcal{I}_k = \{i : y_i = k\}$ index the images with count k. We compute each class centroid and the mean Euclidean radius within that class as

$$\mu_k = \frac{1}{|\mathcal{I}_k|}\sum_{i\in\mathcal{I}_k} z_i, \qquad \sigma_k = \frac{1}{|\mathcal{I}_k|}\sum_{i\in\mathcal{I}_k} \|z_i - \mu_k\|_2. \tag{B.23}$$

The reported score averages the normalized centroid distances over the nine adjacent count pairs:

$$\mathrm{ClassSep} = \frac{1}{9}\sum_{k=1}^{9}\frac{\|\mu_{k+1} - \mu_k\|_2}{\sigma_k + \sigma_{k+1} + 10^{-8}}. \tag{B.24}$$

A larger score indicates greater separation between adjacent count classes relative to the spread within each class. All distances are computed in the original embedding space before t-SNE projection.

B.6 Parameter Count

Table 5 summarizes the parameter budget: ConvStack adds 2.708M parameters and trains 42.827M parameters including the existing visual merger.

Table 5: **Parameter budget of ConvStack-8B.** Percentages are relative to Qwen3-VL-8B's 8.767B parameters, including vision and language components. The shared DeepStack adapter is counted once.

Component	Parameters (M)	% of base model
Vision layer adapter	0.596	0.007
Shared DeepStack adapter	2.112	0.024
Total added by ConvStack	**2.708**	**0.031**
Existing visual merger	40.119	0.458
Total trainable	**42.827**	**0.488**

C Additional Ablation Study

C.1 ViT Injection Layer

We vary the ConvStack injection layer in Qwen3-VL-8B from 0 to 26, retaining adapters at all native DeepStack levels. We select layer 18 based on accuracy on the **PixMo-Count validation split**. Figure 9 reports accuracy on PixMo-Count test and SpatialEval, where the selected layer also achieves the highest accuracy (75.14% and 66.67%, respectively). On PixMo-Count test, it exceeds the runner up, layer 17, by 1.71 percentage points.

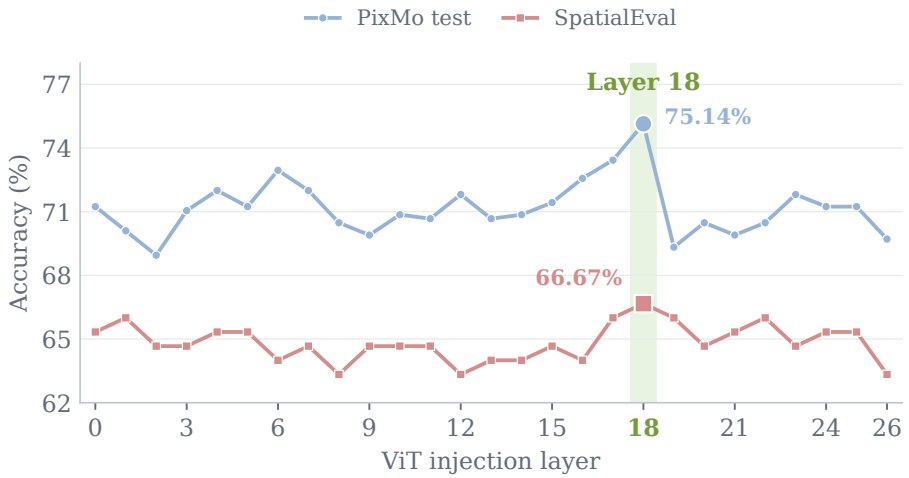


Figure 9: **Ablation of the ViT injection layer.** Accuracy on PixMo-Count test and SpatialEval for Qwen3-VL-8B with ConvStack at each vision layer. The green band marks layer 18, selected using PixMo-Count validation accuracy.

D Quantitative Analysis of Count Representations

We compare Qwen3-VL-8B and ConvStack on the same 500 SynPoly images used in Figure 7, with 50 images per count from 1 to 10. Both analyses use the final decoder state of the last prompt token before the final RMSNorm, computed from the image and counting prompt. We measure separation in this original hidden space.

D.1 Silhouette Across Count Classes

Using cosine distance, let a_i be the mean distance from image i to other images of the same count, and let b_i be the smallest mean distance to any other count class. The silhouette of image i is

$$s_i = \frac{b_i - a_i}{\max(a_i, b_i)}. \tag{D.1}$$

Figure 10(a) reports the mean silhouette for each count. ConvStack improves the score in every class, raising the overall mean from 0.479 to 0.707. Recomputing the metric within counts 6 to 10 gives an increase from 0.319 to 0.627, showing clearer separation among higher counts.

D.2 Separation Between Adjacent Counts

To measure how well neighboring counts are distinguished, we normalize each hidden vector to unit ℓ_2 norm and compute the class centroid μ_n and mean distance to that centroid σ_n, as in Equation (B.23). We report the centroid distance relative to the spread of the two classes:

$$\Gamma_n = \frac{\|\mu_{n+1} - \mu_n\|_2}{\sigma_n + \sigma_{n+1}}, \qquad n \in \{1, \ldots, 9\}. \tag{D.2}$$

As shown in Figure 10(b), ConvStack improves separation for all nine adjacent pairs. The mean Γ_n increases from 0.996 to 1.449. For counts 9 and 10, it increases from 0.558 to 1.058. Together, these analyses show that the improved count clusters in Figure 7 are accompanied by greater separation in the original representation space.

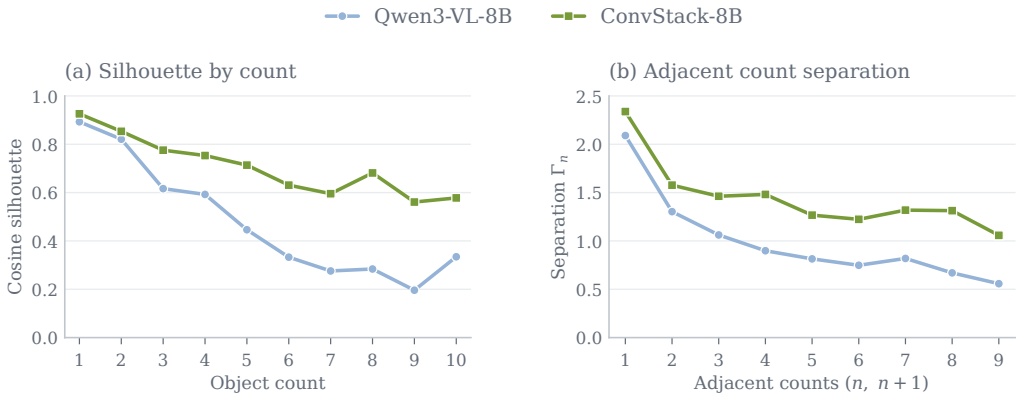


Figure 10: **Quantitative count separation on SynPoly.** (a) Mean cosine silhouette for each of the ten count classes. (b) Separation between counts n and $n+1$, relative to their combined spread. Both models are evaluated on the same images. Larger values indicate better separation.