Can Current AI Models Count What We Mean, Not What They See? A Benchmark and Systematic Evaluation

Gia Khanh Nguyen[1] Yifeng Huang[2] Minh Hoai[1]

[1]Australian Institute for Machine Learning, University of Adelaide, SA, Australia
[2]Stony Brook University, Stony Brook, NY, USA

Abstract

Visual counting is a fundamental yet challenging task, especially when users need to count objects of a specific type in complex scenes. While recent models, including class-agnostic counting models and large vision-language models (VLMs), show promise in counting tasks, their ability to perform fine-grained, intent-driven counting remains unclear. In this paper, we introduce PairTally, a benchmark dataset specifically designed to evaluate fine-grained visual counting. Each of the 681 high-resolution images in PairTally contains two object categories, requiring models to distinguish and count based on subtle differences in shape, size, color, or semantics. The dataset includes both inter-category (distinct categories) and intra-category (closely related subcategories) settings, making it suitable for rigorous evaluation of selective counting capabilities. We benchmark a variety of state-of-the-art models, including exemplar-based methods, language-prompted models, and large VLMs. Our results show that despite recent advances, current models struggle to reliably count what users intend, especially in fine-grained and visually ambiguous cases. PairTally provides a new foundation for diagnosing and improving fine-grained visual counting systems. The data and code for this benchmark are available at https://github.com/bbvisual/PairTally_Benchmark.

1. Introduction

Visual counting is needed in a wide range of situations, from tallying boxes for inventory management and counting cells for disease diagnostics. With the advent of large vision-language models (VLMs)—capable of detecting and recognizing objects and interpreting complex visual scenes—it is natural to explore whether such tasks can now be delegated to machines, particularly in scenarios where multiple object categories are present and models must infer what the user intends to count.

To answer that question, we must first consider how human users might communicate their counting intent to AI models. This is typically done through categorical naming, spatial delineation, or language-based reference. While some of these forms, such as exemplar-based bounding boxes, have been supported by traditional visual counting methods developed before the rise of VLMs, others, particularly natural language instructions and multi-modal integration, are only feasible with more recent advances. Earlier approaches included class-specific models [23, 32, 20, 47, 50, 16, 5, 28, 46, 29, 24, 42, 26, 22, 39, 41, 1] designed for particular object categories such as people, cars, or cells. Later, class-agnostic methods [30, 40, 52, 43, 34, 38, 17] have been developed to count arbitrary objects of interest, given a few exemplars delineated by bounding boxes. The advent of foundation models and large VLMs has significantly enhanced communication flexibility, enabling richer ways to specify counting targets through natural language, visual cues, or a combination of both. Modern counting models [3, 9] can now process an input image along with diverse prompts to identify the objects to be counted, returning a predicted total. VLMs can even support multi-turn conversations, allowing users to provide feedback and request a recount.

While communication with AI models has become more flexible, it is unclear how much their counting accuracy has improved. Do models truly attend to the user's specified target, or do they default to counting the most visually dominant categories, as reinforced by existing datasets and benchmarks? Moreover, can they distinguish targets based on fine-grained attributes such as shape, size, color, or subtle appearance differences?

In this paper, we benchmark ten state-of-the-art visual counting models, organized by their prompting mechanisms and underlying architectures. Our selection comprises four class-agnostic counters—FamNet [40], DAVE [37], GeCo [36], and LoCA [10], which regress density maps from bounding-box exemplars (although DAVE support text prompts through CLIP, we use DAVE exclusively in its bounding-box mode, where it performs best). Next are two

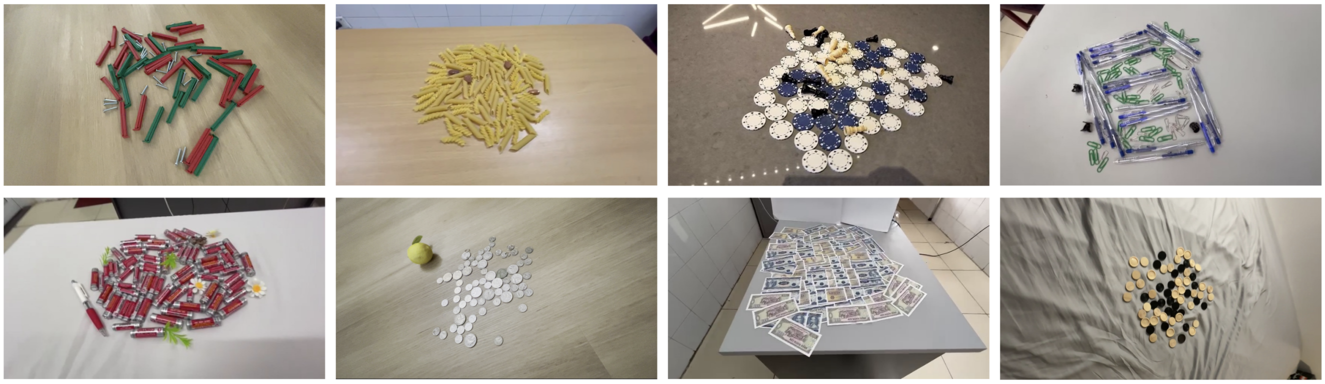

Figure 1. Example samples from the PairTally dataset. Each panel shows intra- and inter-category scenes with fine-grained object variants.

object detectors: CountGD [3], built on Grounding DINO and fine-tuned for counting with both text and bounding-box prompts, and LLMDet [12], a text-only object detector not trained specifically for counting. Finally, we evaluate four large vision–language models, Ovis2 [31], Qwen2.5-VL [4], InternVL3 [56], and LLaMA-3.2 [11], which perform counting via multimodal instruction following. Together, these ten methods span modern counting paradigms, from dense regression to flexible, promptable detectors and instruction-tuned foundation models.

To evaluate these models thoroughly, we need a dataset in which each image contains a substantial number of objects from at least one category, justifying the need for counting, while also including other objects that could plausibly be mistaken for the target category, as often occurs in real-world settings. Unfortunately, existing datasets either feature low object counts across all categories or include many objects predominantly from a single category. Such datasets are inadequate for evaluating fine-grained counting under realistic, challenging conditions.

To address this gap, we introduce **PairTally**, a diagnostic dataset for fine-grained counting. PairTally contains 681 high-resolution images, each featuring two object categories with substantial instance counts (Fig. 1). About half of the pairs involve clearly distinct categories, while the rest contrast subtle subcategories varying in shape, size, or color. Scenes are drawn mainly from tabletop and household contexts for controlled and repeatable setups. The aim is not exhaustive coverage, but targeted analysis of failure modes likely to arise in broader, open-world scenarios.

We use PairTally to examine key challenges in fine-grained visual counting: (1) Do models follow counting prompts reliably? (2) Can they distinguish two object categories in the same scene, whether distinct or subtle variants of the same class? (3) How sensitive are they to fine-grained attributes such as color, size, and texture/shape? This study provides the first systematic evaluation of these challenges in open-world fine-grained counting.

Overall, two consistent patterns emerge. Specialist models tend to overcount, drifting toward counting everything because they indiscriminately enumerate repeated patterns. In contrast, VLMs tend to undercount, as they were not trained for counting but rather for general detection and recognition tasks. Together, these biases show that current methods fall short in fine-grained counting, necessitating models that can both count accurately and distinguish visually similar objects.

2. Related Work

Benchmark Datasets. Early progress in object counting has been shaped by domain-specific datasets such as CARPK [15], NDISPark [8], TRANCOS [13], ShanghaiTech [54], JHU-CROWD++ [44], and VGG-Cell [49]. While effective within narrow applications—vehicles, crowds, or cells—these datasets are unsuitable for evaluating general-purpose counting in open-world, multi-category scenes. FSC-147 [40] and FSCD-147 [34] introduced broader category diversity and remains the dominant benchmark for class-agnostic counting. However, each image of this dataset contains mostly object instances from a single object class, leaving multiclass and fine-grained counting effectively untested.

Recent datasets OmniCount-191 [33], CountBench [35], FSCD-LVIS [34], and MCAC [14] push beyond FSC-147 by introducing greater category diversity, bounding box annotations, and multi-object scenes. However, none offer controlled, real-world images that (i) require counting two object types simultaneously and (ii) demand discrimination between visually similar subtypes. While OmniCount-191 supports multiclass annotations, its coarse grouping and unstructured image compositions fall short of the systematic two-object tests that PairTally delivers.

Benchmark studies. While the literature contains surveys of class-agnostic counting methods, most comprehensively Ciampi*et al.* [6], there is surprisingly little empirical work

Supercategory	Category (Subcategory 1, Subcategory 2)
Food	pasta (spiral, penne), lime (citrus, calamansi), peppercorn (black, white), tomato (normal, baby), chili (long, short), peanut (with/without skin), bean (black, soy), seed (pumpkin, sunflower), coffee candy (brown, black), garlic, shallot
Fun	checker piece (black, white), mahjong tile (bamboo, character), lego piece (green, pink), chess piece (black, white), puzzle piece (edge, center), puzzle piece (edge, center), poker chip (blue, white), playing card (red, black), marble (big, small), dice (green, white), Chinese card (red, black)
Household	toothpick (straight, plastic), cotton bud (wooden, plastic), pill (white, yellow), battery (AAA, AA), hair clipper (black, brown), bill (1000, 5000 VND), coin (5¢, 10¢), bottle cap (beer, plastic), shirt button (2, 4 holes), utensil (spoon, fork)
Office	push pin (normal, round), heart sticker (big, small), craft stick (red/orange, blue/purple), rubber band (yellow, blue), sticky note (green shades), paper clip (silver, colored), pen (with/without cap), pencil, rhinestone (round, star), zip tie (short, long), safety pin (big, small), lapel pin, wire organizer
Other	screw (bronze, silver), bolt (hex, mushroom), nut (hex, square), washer (metal, nylon), bead (blue/pink shades), ikea clip (green, red), peg (grey, white), stone (red, yellow), novelty buttons (beige, transparent)

Table 1. Object categories of the PairTally dataset. Subcategory division are based on: color, size, texture/shape, no subcategories.

that systematically dissects what current models can and cannot count. The recent PrACo benchmark [7] makes a valuable contribution by introducing tests for prompt sensitivity. However, despite this advance, PrACo falls short of testing a model's ability to perform fine-grained and multi-class counting under realistic conditions. Its mosaic test images are synthetically composed by stitching together crops of single-category scenes, meaning models are never required to interpret two object types in the same physical context or deal with natural clutter, occlusion, or layout. As mentioned above due to a lack of multi-class datasets, there is also a lack of empirical studies that assess fine-grained counting abilities. Our study is the first systematic, cross-paradigm audit under tests that simultaneously demand (i) distinguishing different object types and (ii) discriminating subtle intra-category differences (color, size, shape); our analysis thus fills a critical gap between broad surveys and ad-hoc model demonstrations.

3. State of the art counting models

Counting models that use visual exemplars. We benchmark four counting models from this category: FamNet [40], LOCA [10], GeCo [36], and DAVE [37]. These follow the exemplar-guided paradigm, where one or more visual exemplars are provided at test time, and the model must count similar objects in a query image.

We do not benchmark several other exemplar-based methods. CACViT [48], ConCoNet [45], SAFECount [53], and BMNet+ [43] either do not achieve state-of-the-art results or offer limited novelty over baselines being tested. Our selected models are representative, influential, and well-suited to the exemplar-guided setting we evaluate.

Counting models that use language prompts Prompt-based and open-world counting methods aim to generalize beyond fixed category sets, typically without relying on explicit visual exemplars. These models leverage natural language prompts to guide object localization and counting, enabling flexible and open-ended reasoning.

We benchmark CountGD and LLMDet, which represent a shift from the exemplar-based approach to the prompt-compatible counting approach. Both use open-vocabulary detectors to predict bounding boxes based on textual descriptions, bypassing the need for reference crops. CountGD builds on GroundingDINO [27], while LLMDet [12] is a newer object detection model that achieves improved accuracy and grounding precision via language-guided detection and counting.

Other text-based methods such as CLIP-Count [19], VL-Counter [21], CounTX [2], ZSC [51], VA-Count [55], PseCo [18] and TFPOC [25] are not included due to limited performance on recent benchmarks. GroundingREC [9] is excluded as its referring expression comprehension design does not transfer well to our dataset format.

Counting with Vision-Language Models. Recent VLMs such as Ovis2 [31], Qwen2.5-VL [4], LLaMA-3.2 [11], and InternVL3 [56] represent a new generation of multimodal systems that combine large language models with visual encoders. These models are commonly used for tasks such as image captioning, visual question answering, and referring expression comprehension, where they generate text responses grounded in visual input. To evaluate their capabilities in fine-grained visual counting, we selected mid- to smaller-sized variants—Ovis2 (16B), Qwen2.5-VL (7B), LLaMA-3.2 (11B), and InternVL3 (14B)—rather than their largest counterparts. This decision was driven by practical considerations around resource constraints, and accessibility. By focusing on more deployable model sizes, we aim to provide insights that are both scalable and relevant to real-world use cases, while still capturing the core strengths and limitations of these architectures.

We do not benchmark commercial models (e.g., ChatGPT, Gemini) due to their proprietary nature, which hinders reproducibility. Our findings therefore focus on open-source VLMs. While we make no formal claims about commercial models, we believe they are unlikely to substantially outperform those tested, based on known performance gaps and limited preliminary experiments.

4. PairTally – a New Benchmark Dataset

Pair Construction and Scene Design. PairTally is organized into five broad *supercategories*: Food, Fun, Household, Office, and Other. The dataset comprises 54 object *categories*. For each category (e.g. *poker chips*, *pens*), we manually defined up to two visually distinct *subcategories* based on attributes of interest: color (43.5% of subcategory distinctions), texture or shape (42.5%), and size (14.1%). This was done in a way that ensure real-world relevance simulating scenarios that a human would go through such as counting game pieces or sorting office items. We prioritized these real life examples to capture the kinds of visual variation that would occur in daily life. This yielded 98 subcategories in total as shown in Table 1.

We next systematically constructed an initial set of 100 *subcategory pairs*: 50 **intra-category** pairs that match the two subcategories within each category, and 50 **inter-category** pairs chosen across categories. Later on, we excluded three intra-category pairs due to insufficient visual distinction (e.g., "light green vs. mint green cups"). Ambiguities also emerged when each subclass contained more than three fine-grained variations, yet some models allowed only three exemplar bounding boxes per query. For instance, in "small vs. big M&Ms," both sizes appeared in various colors (red, yellow, green, blue), making it unclear whether the model should count all items of a given size or only those matching the color of the exemplars. To ensure consistent interpretation and fair evaluation, we excluded such ambiguous cases from the benchmark. The final set comprises 97 subcategory pairs.

Data capture and annotation. All scenes were recorded using consumer-grade smartphones (iPhone 12, Samsung S21 Plus), capturing natural diversity in lighting, perspective, occlusion, and scene complexity. Each image was annotated with the following: **three bounding boxes** each (for exemplar based counting), **textual labels** identifying object categories and fine-grained variants (for prompt based counting), **count labels** denoting the number of instances per object type in each scene.

5. Experiments

For all evaluations, each model was asked to count the same things, differing only in how exemplars and text prompts were supplied. DAVE, GeCo, LoCA, and FamNet received three exemplar bounding boxes per scene; CountGD was evaluated in two modes—exemplar + text (three exemplar boxes and a textual reference) and text-only; LLMDet operated in text-only mode (textual reference); and Vision–Language Models (Ovis2, Qwen2.5-VL, LLaMA-3.2, InternVL3) were queried with the unified prompt: "Count the number of {object} in this image. Provide only the total count in this format: <count>N</count>. If you see no {object} or are unsure, respond with <count>0</count>." The placeholder {object}" was replaced with the specific reference (e.g., red poker chips"), and counts were extracted by parsing the integer within the <count>. . . </count> tags.

5.1. Main results

Recall that each image in our benchmark dataset contains objects from two categories, along with a few possible distractor objects from other classes. The two main categories are unordered, but for convenience, we refer to them as A and B, with a and b denoting the number of objects in each category, respectively. For a given method, let $f(A)$ represent the predicted count when the model is prompted to count objects in category A, and let $f(A+B)$ denote the predicted count when the model is asked to count both A and B. In the latter case, if the method requires bounding-box exemplars, we provide two from A and one from B.

To evaluate the accuracy of each method, we use the absolute difference between the predicted count and the ground truth. Specifically, we compute $|f(A) - a|$ and $|f(A+B) - (a+b)|$ to assess how well the predicted counts match the expected values for each prompt.

To further analyze model behavior, we also compute $|f(A) - (a+b)|$ and compare it with $|f(A) - a|$ to evaluate whether the model mistakenly counts objects from category B when only A is requested. Additionally, we compute $|f(A) - f(A+B)|$ to assess whether the model is responsive to the prompt—that is, whether it adjusts its predictions based on the requested categories or tends to produce similar outputs regardless of the prompt content.

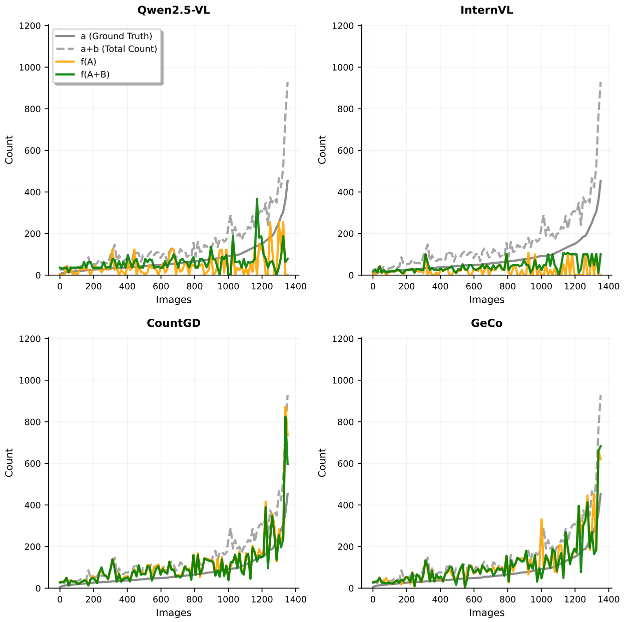

Figure 2. Model counting predictions sorted by increasing ground truth count. Solid grey: ground truth queried category a. Dashed grey: total count a+b. Orange: predicted count f(A) when querying single category. Green: predicted count f(A+B) when querying both categories.

The results in Figure 2 and Table 2 reveal distinct failure modes across model categories. Vision-Language Models

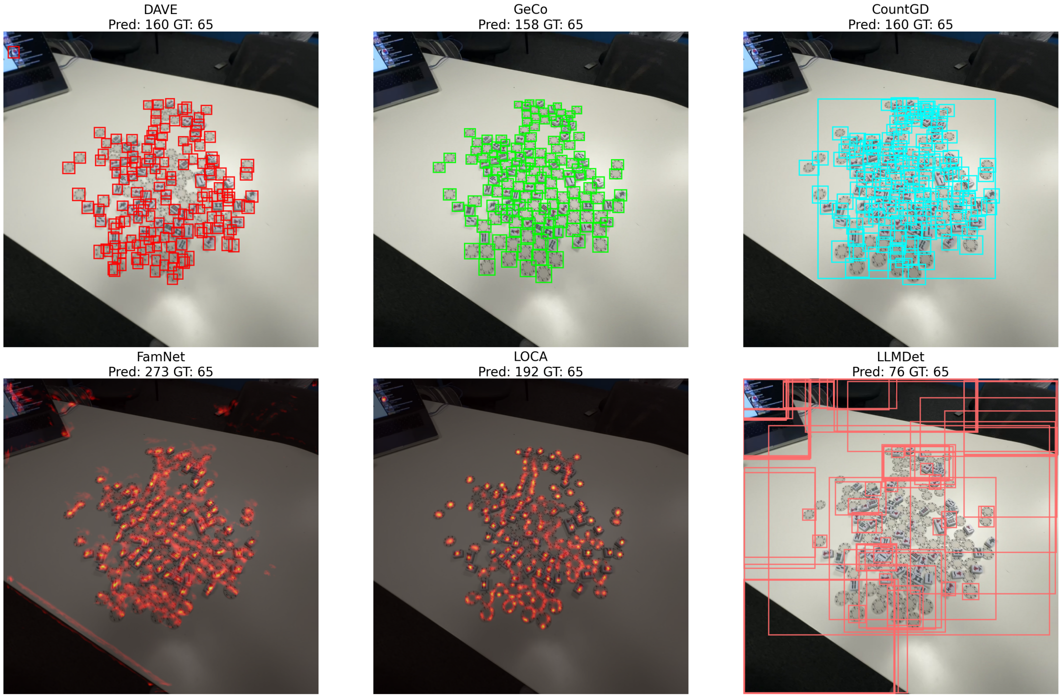

Figure 3. Inter-category scene with Mahjong tiles and poker chips; models were asked to count only Mahjong tiles.

		Mean over test images				Percentage over test images	
Model	Prompt	$\lvert f(A)-a\rvert$	$\lvert f(A+B)-(a+b)\rvert$	$\lvert f(A)-(a+b)\rvert$	$\lvert f(A)-f(A+B)\rvert$	$f(A)>a$	$\lvert f(A)-a\rvert > \lvert f(A)-(a+b)\rvert$
DAVE [37]	b.boxes	47.37	69.49	69.89	18.00	74.0	48.6
GeCo [36]	b.boxes	50.24	53.07	51.81	21.17	84.7	59.8
LOCA [10]	b.boxes	72.80	62.78	76.51	39.73	83.1	58.6
FamNet [40]	b.boxes	75.83	88.30	90.60	51.57	73.1	55.7
Count GD [3]	both	46.67	57.33	52.88	10.58	83.1	57.3
Count GD (Text) [3]	text	50.32	57.33	93.99	42.83	52.0	32.9
LLMDet [12]	text	77.23	107.84	121.82	45.49	53.3	41.4
Ovis2 [31]	text	65.15	111.56	145.37	35.69	5.0	0.3
Qwen2.5-VL [4]	text	59.36	99.88	126.88	43.62	26.1	10.7
LLaMA-3.2 [11]	text	54.67	97.56	130.46	45.15	12.9	2.5
InternVL3 [56]	text	63.41	115.98	142.10	27.59	9.1	1.0

Table 2. Fine-grained counting results comparing single-category $f(A)$ with dual-category $f(A+B)$ against ground truths a and b.

(e.g., Qwen2.5-VL, InternVL) exhibit erratic and inconsistent predictions that often bear little correlation to either the ground truth or total counts. Their outputs fluctuate unpredictably across images, suggesting fundamental difficulties in the visual counting task itself, with predictions largely disconnected from the true object quantities.

By contrast, specialist counting models (e.g., CountGD, GeCo) display a more systematic but equally problematic behavior: overcounting. When prompted with a single category A, these models frequently predict $f(A) > a$, as shown in Table 2, where over 70% of cases for DAVE and FamNet and higher rates for GeCo and CountGD exceed the true count. Moreover, for GeCo and CountGD, $|f(A) - a| > |f(A) - (a + b)|$ more than half of the time, indicating that their predictions drift toward the total object count rather than the queried subset. In other words, these models perceive object quantities but systematically conflate A and B, counting both even when only A is requested.

Figure 2 illustrates this contrast. The orange curves ($f(A)$) of specialist models often track the dashed line ($a + b$) more closely than the solid line (a), a clear sign of category confusion and overcounting. Conversely, their green curves ($f(A + B)$) align more closely with $a + b$, showing that they are more reliable when tasked with total counts rather than fine-grained ones. VLMs, on the other hand, show no such consistent structure, underscoring their weakness in the basic counting task itself.

Overall, these results suggest that while VLMs fail at robust counting altogether, specialist models succeed at estimating totals but struggle to follow prompts for fine-grained counting. This indicates that their limitation lies not in visual perception but in instruction-following, whereas VLMs face the challenge of weak visual counting overall. The contrasting failure modes point to fundamentally different architectural bottlenecks that must be addressed for reliable fine-grained counting.

Fig. 3 shows some representative results of methods that output detections or density maps. GeCo and CountGD locate objects well but confuse similar types, leading to overcounts. DAVE sometimes separates categories but is prone to background false positives or double-counting fragmented objects. LLMDet, despite its general vision capabilities, underperforms—likely due to limited training on dense scenes. LOCA consistently overcounts, and FamNet fails to isolate relevant objects altogether.

5.2. Performance on inter- and intra- scenes

In addition to using the MAE metric, we also use *Normalized Absolute Error (NAE)*, a scale-aware version of MAE that adjusts for the number of objects in each image: NAE $= \frac{1}{N}\sum_{i=1}^{N}\frac{|\hat{y}_i - y_i|}{y_i}$. This enables fairer comparison across scenes with varying object counts. As shown in Table 3, we compare NAE between inter-class scenes (objects from different categories) and intra-class scenes (objects from the same or visually similar categories). Relative changes are reported as percentage increases or decreases when moving from inter-class to intra-class settings. Among learned models, DAVE and CountGD (Text) are the most robust, with NAE increases of only 2.3% and 18.1%. LOCA surprisingly improves in intra-class settings, reducing its NAE by 19.3%. By contrast, CountGD and GeCo show large NAE increases (+34.6% and +20.3%) in intra-class settings, indicating higher distractor sensitivity. FamNet and LLMDet also degrade (+5.6% and +60.3%).

VLMs show relatively stable NAE but high absolute errors. This stems from a tendency to default to mid- to high-range guesses (e.g., 36, 63, 67), regardless of the true count. For instance, Qwen2.5-VL often predicts counts between 30–100. Though incorrect, these guesses yield lower NAE in high-count scenes due to normalization. This behavior suggests that vision–language models may rely more on priors or prompt biases than genuine enumeration.

To understand this further, we experimented with several prompting strategies, including negation prompts that explicitly instructed models to count only one object type and *not* the other. We also applied prompt engineering with more detailed descriptions and provided bounding box coordinates as reference. However, none of these techniques improved performance; in fact, they often degraded accuracy or led to further instability in count predictions.

Model	MAE ↓		NAE ↓	
	Inter	Intra	Inter	Intra
Mean Baseline	39.42	66.71	0.714	0.535
Median Baseline	37.38	57.25	1.587	0.776
DAVE [37]	46.27	46.75	0.779	0.797
GeCo [36]	45.05	54.80	0.777	0.935
LoCA [10]	71.89	57.45	1.177	0.950
FamNet [40]	66.97	74.75	1.363	1.440
Count GD [3]	39.78	56.54	0.673	0.906
Count GD (Text) [3]	50.23	53.93	0.712	0.841
LLMDet [12]	78.72	142.08	0.661	1.060
Ovis2 [31]	56.87	74.24	0.711	0.736
Qwen2.5-VL [4]	46.35	67.86	0.598	0.712
LLaMA-3.2 [11]	49.14	58.73	0.730	0.740
InternVL3 [56]	55.89	71.47	0.667	0.721

Table 3. MAE and NAE across inter- and intra-scenes (lower is better).

5.3. Most and least sensitive visual attributes

Table 4 reports model performance on three subsets of the intra-category test images, divided based on how the two object subcategories in each image differ—specifically by color, size, or texture/shape. Each subset contains images where the objects differ only by the corresponding attribute. We report MAE, RMSE, and NAE, but NAE is preferred for cross-subset comparison as it normalizes object counts, making the comparison more reliable.

Color-based variation produces the lowest NAEs overall. DAVE (0.738) and CountGD-Text (0.760) outperform their own results on size and texture, suggesting hue is a more reliable cue than expected. GeCo also achieves low error (0.791), reinforcing that color is the easiest attribute for models to exploit.

Size variation, by contrast, yields consistently higher NAEs (1.24–1.41). This challenges the intuition that spatial differences aid separation: scale changes complicate localization, particularly when small objects cluster or large ones occlude others. GeCo's 1.345 again indicates reliance on global rather than local features.

Texture/shape variation falls in between. CountGD performs best (0.735), while GeCo (0.946) and FamNet (1.448) are less robust, likely due to clutter and background interference.

In sum, models leverage color most effectively, while size and texture distinctions remain challenging—paralleling human difficulty with such attributes. These findings underline the need for stronger feature disentanglement and attribute-specific supervision in fine-grained counting models.

Model	**Evaluation on intra-category subsets (objects differ by)**								
	Color			**Size**			**Texture/Shape**		
	MAE ↓	RMSE ↓	NAE ↓	MAE ↓	RMSE ↓	NAE ↓	MAE ↓	RMSE ↓	NAE ↓
Mean Baseline	55.16	83.51	1.698	25.81	36.14	**1.154**	43.75	59.38	0.838
Median Baseline	49.37	88.01	0.967	24.71	36.75	**0.586**	40.89	61.71	0.720
DAVE [37]	63.44	89.16	**0.738**	33.26	39.31	1.293	34.14	43.00	0.693
GeCo [36]	63.40	88.77	**0.791**	35.06	41.13	1.345	52.53	74.55	0.946
LoCA [10]	65.37	95.24	**0.799**	33.34	39.66	1.244	57.33	91.11	1.007
FamNet [40]	84.92	117.33	**1.296**	56.32	75.44	1.859	70.45	90.64	1.448
CountGD [3]	75.40	117.32	**0.856**	36.30	42.35	1.402	43.95	57.42	0.793
CountGD (Text) [3]	64.51	98.99	**0.760**	38.95	45.61	1.410	48.06	73.20	0.735
LLMDet [12]	118.29	151.01	**2.12**	68.68	82.73	4.18	89.33	116.06	2.04

Table 4. Performance comparison on intra-category subsets (color, size, texture/shape).

6. Conclusions

This study presents a targeted, cross-paradigm evaluation of fine-grained counting using a diagnostic benchmark, highlighting the limitations of counters, promptable detectors, and open-source mid-sized vision-language models when precision and intent sensitivity are required. As part of this effort, we introduce PairTally, a benchmark dataset containing images with two object categories, including both inter-category and intra-category pairs. PairTally is designed to test a model's ability to count selectively in visually complex scenes. Our evaluation of ten state-of-the-art models reveals that they often fail when subtle visual distinctions are necessary—such as distinguishing pill types, identifying species, or counting similar tools—where fine-grained accuracy is critical. These findings underscore the need for more capable models, as well as more suitable training data to support their development.

Acknowledgment. This work was funded by the Australian Institute for Machine Learning (University of Adelaide) and the Centre for Augmented Reasoning, an initiative by the Department of Education, Australian Government. The authors would also like to thank Viet Bach Tran for assistance with data collection.

References

[1] S. Abousamra, M. Hoai, D. Samaras, and C. Chen. Localization in the crowd with topological constraints. In *Proc. AAAI*, 2021.

[2] N. Amini-Naieni, K. Amini-Naieni, T. Han, and A. Zisserman. Open-world text-specified object counting. *arXiv:2306.01851*, 2023.

[3] N. Amini-Naieni, T. Han, and A. Zisserman. Countgd: Multi-modal open-world counting. *NeurIPS*, 2024.

[4] S. Bai, K. Chen, X. Liu, J. Wang, W. Ge, S. Song, K. Dang, P. Wang, S. Wang, J. Tang, et al. Qwen2. 5-vl technical report. *arXiv:2502.13923*, 2025.

[5] S. Bai, Z. He, Y. Qiao, H. Hu, W. Wu, and J. Yan. Adaptive dilated network with self-correction supervision for counting. In *CVPR*, 2020.

[6] L. Ciampi, A. Azmoudeh, E. E. Akbaba, E. Saritacs, Z. A. Yazici, H. K. Ekenel, G. Amato, and F. Falchi. A survey on class-agnostic counting: Advancements from reference-based to open-world text-guided approaches. *arXiv:2501.19184*, 2025.

[7] L. Ciampi, N. Messina, M. Pierucci, G. Amato, M. Avvenuti, and F. Falchi. Mind the prompt: A novel benchmark for prompt-based class-agnostic counting. In *WACV*, 2025.

[8] L. Ciampi, C. Santiago, J. P. Costeira, C. Gennaro, and G. Amato. Domain adaptation for traffic density estimation. In *VISIGRAPP (5: VISAPP)*, 2021.

[9] S. Dai, J. Liu, and N.-M. Cheung. Referring expression counting. In *CVPR*, 2024.

[10] N. DJukić, A. Lukevzivc, V. Zavrtanik, and M. Kristan. A low-shot object counting network with iterative prototype adaptation. In *CVPR*, 2023.

[11] A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan, et al. The llama 3 herd of models. *arXiv–2407*, 2024.

[12] S. Fu, Q. Yang, Q. Mo, J. Yan, X. Wei, J. Meng, X. Xie, and W.-S. Zheng. Llmdet: Learning strong open-vocabulary object detectors under the supervision of large language models. In *CVPR*, 2025.

[13] R. Guerrero-Gómez-Olmedo, B. Torre-Jiménez, R. López-Sastre, S. Maldonado-Bascón, and D. Onoro-Rubio. Extremely overlapping vehicle counting. In *Iberian conference on pattern recognition and image analysis*, 2015.

[14] M. Hobley and V. Prisacariu. Abc easy as 123: A blind counter for exemplar-free multi-class class-agnostic counting. In *ECCV*, 2024.

[15] M.-R. Hsieh, Y.-L. Lin, and W. H. Hsu. Drone-based object counting by spatially regularized regional proposal network. In *ICCV*, 2017.

[16] Y. Hu, X. Jiang, X. Liu, B. Zhang, J. Han, X. Cao, and D. Doermann. Nas-count: Counting-by-density with neural architecture search. In *ECCV*, 2020.

[17] Y. Huang, V. Ranjan, and M. Hoai. Interactive class-agnostic object counting. In *Proc. ICCV*, 2023.

[18] Z. Huang, M. Dai, Y. Zhang, J. Zhang, and H. Shan. Point segment and count: A generalized framework for object counting. In *CVPR*, 2024.

[19] R. Jiang, L. Liu, and C. Chen. Clip-count: Towards text-guided zero-shot object counting. In *Proceedings of the 31st ACM International Conference on Multimedia*, 2023.

[20] X. Jiang, L. Zhang, M. Xu, T. Zhang, P. Lv, B. Zhou, X. Yang, and Y. Pang. Attention scaling for crowd counting. In *CVPR*, 2020.

[21] S. Kang, W. Moon, E. Kim, and J.-P. Heo. Vlcounter: Text-aware visual representation for zero-shot object counting. In *AAAI*, 2024.

[22] I. H. Laradji, N. Rostamzadeh, P. O. Pinheiro, D. Vazquez, and M. Schmidt. Where are the blobs: Counting by localization with point supervision. In *ECCV*, 2018.

[23] Y. Li, X. Zhang, and D. Chen. Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In *CVPR*, 2018.

[24] D. Lian, J. Li, J. Zheng, W. Luo, and S. Gao. Density map regression guided detection network for rgb-d crowd counting and localization. In *CVPR*, 2019.

[25] W. Lin and A. B. Chan. A fixed-point approach to unified prompt-based counting. In *AAAI*, 2024.

[26] C. Liu, X. Weng, and Y. Mu. Recurrent attentive zooming for joint crowd counting & precise localization. In *CVPR*, 2019.

[27] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, Q. Jiang, C. Li, J. Yang, H. Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *ECCV*, 2024.

[28] X. Liu, J. Yang, W. Ding, T. Wang, Z. Wang, and J. Xiong. Adaptive mixture regression network with local counting map for crowd counting. In *ECCV*, 2020.

[29] Y. Liu, M. Shi, Q. Zhao, and X. Wang. Point in, box out: Beyond counting persons in crowds. In *CVPR*, 2019.

[30] E. Lu, W. Xie, and A. Zisserman. Class-agnostic counting. In *ACCV*, 2018.

[31] S. Lu, Y. Li, Q.-G. Chen, Z. Xu, W. Luo, K. Zhang, and H.-J. Ye. Ovis: Structural embedding alignment for multimodal large language model. *arXiv:2405.20797*, 2024.

[32] Y. Miao, Z. Lin, G. Ding, and J. Han. Shallow feature based dense attention network for crowd counting. In *AAAI*, 2020.

[33] A. Mondal, S. Nag, X. Zhu, and A. Dutta. Omnicount: Multi-label object counting with semantic-geometric priors. In *AAAI*, 2025.

[34] T. Nguyen, C. Pham, K. Nguyen, and M. Hoai. Few-shot object counting and detection. In *ECCV*, 2022.

[35] R. Paiss, A. Ephrat, O. Tov, S. Zada, I. Mosseri, M. Irani, and T. Dekel. Teaching clip to count to ten. In *CVPR*, 2023.

[36] J. Pelhan, A. Lukezic, V. Zavrtanik, and M. Kristan. A novel unified architecture for low-shot counting by detection and segmentation. *NeurIPS*, 2024.

[37] J. Pelhan, V. Zavrtanik, M. Kristan, et al. Dave-a detect-and-verify paradigm for low-shot counting. In *CVPR*, 2024.

[38] V. Ranjan and M. Hoai. Exemplar free class agnostic counting. In *Proc. ACCV*, 2022.

[39] V. Ranjan, H. Le, and M. Hoai. Iterative crowd counting. In *Proc. ECCV*, 2018.

[40] V. Ranjan, U. Sharma, T. Nguyen, and M. Hoai. Learning to count everything. In *CVPR*, 2021.

[41] V. Ranjan, B. Wang, M. Shah, and M. Hoai. Uncertainty estimation and sample selection for crowd counting. In *Proc. ACCV*, 2020.

[42] D. B. Sam, S. V. Peri, M. N. Sundararaman, A. Kamath, and V. B. Radhakrishnan. Locate, size and count: Accurately resolving people in dense crowds via detection. *TPAMI*, 2020.

[43] M. Shi, H. Lu, C. Feng, C. Liu, and Z. Cao. Represent, compare, and learn: A similarity-aware framework for class-agnostic counting. In *CVPR*, 2022.

[44] V. A. Sindagi, R. Yasarla, and V. M. Patel. Jhu-crowd++: Large-scale crowd counting dataset and a benchmark method. *IEEE transactions on pattern analysis and machine intelligence*, 44(5):2594–2609, 2020.

[45] A. F. O. Soliven, J. J. Virtusio, J. J. M. Ople, D. S. Tan, D. Amalin, and K.-L. Hua. Conconet: Class-agnostic counting with positive and negative exemplars. *Pattern Recognition Letters*, 171:148–154, 2023.

[46] Q. Song, C. Wang, Z. Jiang, Y. Wang, Y. Tai, C. Wang, J. Li, F. Huang, and Y. Wu. Rethinking counting and localization in crowds: A purely point-based framework. In *ICCV*, 2021.

[47] B. Wang, H. Liu, D. Samaras, and M. Hoai. Distribution matching for crowd counting. In *NeurIPS*, 2020.

[48] Z. Wang, L. Xiao, Z. Cao, and H. Lu. Vision transformer off-the-shelf: A surprising baseline for few-shot class-agnostic counting. In *AAAI*, 2024.

[49] W. Xie, J. A. Noble, and A. Zisserman. Microscopy cell counting and detection with fully convolutional regression networks. *Computer methods in biomechanics and biomedical engineering: Imaging & Visualization*, 6(3):283–292, 2018.

[50] C. Xu, K. Qiu, J. Fu, S. Bai, Y. Xu, and X. Bai. Learn to scale: Generating multipolar normalized density maps for crowd counting. In *ICCV*, 2019.

[51] J. Xu, H. Le, V. Nguyen, V. Ranjan, and D. Samaras. Zero-shot object counting. In *CVPR*, 2023.

[52] S.-D. Yang, H.-T. Su, W. H. Hsu, and W.-C. Chen. Class-agnostic few-shot object counting. In *Proc. WACV*, 2021.

[53] Z. You, K. Yang, W. Luo, X. Lu, L. Cui, and X. Le. Few-shot object counting with similarity-aware feature enhancement. In *Proc. WACV*, 2023.

[54] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma. Single-image crowd counting via multi-column convolutional neural network. In *CVPR*, 2016.

[55] H. Zhu, J. Yuan, Z. Yang, Y. Guo, Z. Wang, X. Zhong, and S. He. Zero-shot object counting with good exemplars. In *ECCV*, 2024.

[56] J. Zhu, W. Wang, Z. Chen, Z. Liu, S. Ye, L. Gu, H. Tian, Y. Duan, W. Su, J. Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv:2504.10479*, 2025.