SEE, MEASURE, AND REASON: LEARNING VISUALLY GROUNDED REASONING IN PATHOLOGY

Anonymous authors
Paper under double-blind review

ABSTRACT

Pathological assessment relies on recognizing fine-grained visual details in histological images. Vision-language models (VLMs) increasingly support pathology interpretation, yet their ability to perceive these details remains inadequate. This weakness leads to inaccurate cellular observations that can persist even when final answers are correct. In this paper, we propose ASPECT to improve visually grounded reasoning through explicit supervision of cellular appearance and abundance. ASPECT trains intermediate visual tokens through pathology feature reconstruction, cell feature alignment, and count supervision. Three-stage supervised fine-tuning teaches the model to perceive, generate visual tokens, and reason, followed by reinforcement learning that rewards answer correctness and consistency with reported measurements. We also introduce PathoVernier, a benchmark of 759 expert-reviewed questions from five pathology datasets covering four cellular composition tasks. It evaluates both final answers and intermediate measurements to expose errors hidden by answer accuracy. On PathoVernier, ASPECT achieves relative accuracy gains of approximately 19.2% over the strongest baseline, Gemini-3.1-Pro, and 99.3% over its Qwen3-VL-8B backbone, while reducing RAWR, which measures counting errors within correct responses, by 28.1% and 42.7%, respectively. ASPECT also improves over its backbone on three external pathology benchmarks covering classification and question answering beyond cellular composition tasks.

1 INTRODUCTION

Pathology vision-language models (VLMs) can describe tissue morphology and answer diagnostic questions from histological images (Lu et al., 2024; Seyfioglu et al., 2024; Sun et al., 2025; Zhang et al., 2026c). However, success in question answering does not establish how accurately a model identifies the visual details needed for its answers. Recent pathology evaluations reveal difficulties with localization, counting, and spatial relationships, even on slides where models answer higher-level questions correctly (Chen et al., 2026; Zhang et al., 2026a). These findings motivate a closer examination of the visual understanding underlying pathology VLM predictions.

General-purpose VLM evaluations examine fine-grained perception through spatial judgments and counting (Fu et al., 2024; Rahmanzadehgervi et al., 2024; Zhou et al., 2025). In pathology, cellular composition analysis requires these abilities and supports subsequent tissue assessment. Tumor content informs specimen suitability for molecular testing (Lindeman et al., 2018), while assessment of tumor-infiltrating lymphocytes requires identifying inflammatory cells within the appropriate tissue compartment (Salgado et al., 2015). Distinguishing cell types and comparing their abundance across densely populated regions therefore provide a clinically relevant test of visual understanding. Nucleus instance annotations supply reference identities and positions, allowing both the reported measurements and the conclusions drawn from them to be checked.

Our evaluation reveals errors in both observing cellular composition and using those observations to answer questions. Figure 1 illustrates these failures in GPT-5.5 (OpenAI, 2026a) and Patho-R1 (Zhang et al., 2026c), representing a frontier closed-source VLM and a pathology-tuned model. In Figure 1(a), both models reverse the relative epithelial-like nuclear densities of two regions. In Figure 1(b), both select the correct answer despite inaccurate counts; Patho-R1 additionally selects a category inconsistent with the ratio implied by its own counts. *Thus, a correct answer can hide both*

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

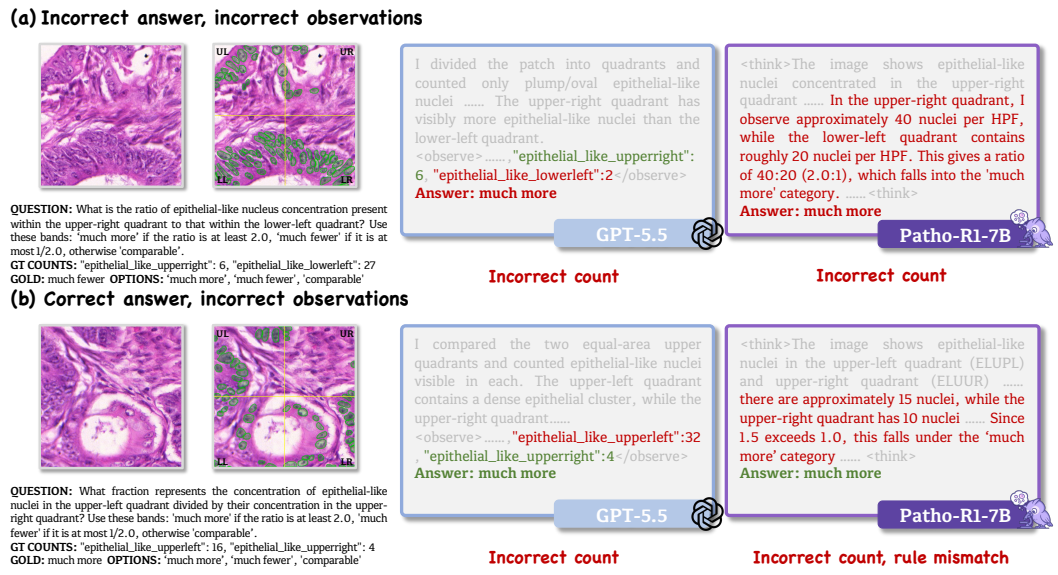


Figure 1: Correct answers can conceal inaccurate observations. GPT-5.5 and Patho-R1 reverse regional nuclear densities in (a) and answer correctly despite incorrect counts in (b), where Patho-R1 also misapplies the ratio rule. Green contours mark reference nucleus annotations.

inaccurate observation and incorrect use of the stated evidence. We hypothesize that fine-tuning focused on answer generation can strengthen pathology-specific language priors without comparable gains in extracting and using visual information. Consistent with this hypothesis, pathology-tuned models retain substantial accuracy without images, and their gains over base models need not coincide with better visual grounding (Zhang et al., 2026a). Similar image-independent performance has been reported on broader medical multimodal benchmarks (Asadi et al., 2026). These findings motivate evaluating the observations behind an answer and training models to make accurate observations and use them correctly.

To connect fine-grained perception with pathology reasoning, we introduce ASPECT and PathoVernier. ASPECT supervises intermediate visual tokens with pathology features and cell counts, using three-stage supervised fine-tuning (SFT) to generate quantitative observations and reason from them. Reinforcement learning (RL) further rewards answer correctness and consistency with the reported counts. PathoVernier provides 759 expert-reviewed questions from five datasets spanning more than 15 organs. Four cellular composition tasks pair each answer with reference counts and a decision rule, allowing evaluation beyond final-answer accuracy. Our main contributions are summarized as follows:

- We introduce PathoVernier to assess fine-grained pathology perception through four cellular composition tasks. Its RAWR metric (right answer, wrong reason) exposes counting errors hidden by correct answers.
- We propose ASPECT, which trains intermediate visual tokens to encode pathology appearance and cell abundance through three-stage SFT, followed by RL for answer correctness and consistency with reported counts.
- We conduct experiments against ten baselines. ASPECT-8B leads on all PathoVernier metrics, improving accuracy by 19.2% and reducing RAWR by 28.1% relative to the strongest baseline. Gains on three external benchmarks extend beyond cellular composition tasks.

2 Related Work

Pathology Vision-Language Models Pathology VLMs combine domain-specific visual representations with language supervision to describe histological findings and answer diagnostic questions (Lu et al., 2024; Seyfioglu et al., 2024; Sun et al., 2025). PathChat combines a pathology vision

encoder with language instruction tuning (Lu et al., 2024). Reasoning-oriented models subsequently introduce chain-of-thought supervision and reinforcement learning (Zhang et al., 2026c). More recently, PathReasoner-R1 uses knowledge-guided reasoning data and entity-based rewards to strengthen the connection between pathological findings and diagnoses (Jiang et al., 2026). PathFound seeks additional evidence to refine diagnoses (Hua et al., 2026). RECAP-PATH optimizes prompts for morphological descriptions using diagnostic feedback (Hong et al., 2026). However, diagnostic accuracy and plausible explanations do not establish whether reported cell counts are correct or support a quantitative conclusion.

Evaluating Visual Understanding Visual-understanding benchmarks assess fine-grained perception, visual reasoning, and domain-specific knowledge (Fu et al., 2024; Sun et al., 2024; Chen et al., 2024b). General-domain evaluations isolate perceptual operations, whereas pathology benchmarks emphasize morphological recognition and diagnostic reasoning. For example, BLINK evaluates abilities such as visual correspondence, relative depth estimation, and multi-view reasoning (Fu et al., 2024), while PathMMU assesses expert-level pathology understanding through question answering (Sun et al., 2024). More targeted pathology evaluations examine image dependence, entity–region correspondence, and visual understanding across local regions and whole-slide views (Chen et al., 2026; Zhang et al., 2026a). However, these pathology evaluations largely assess visual abilities through separate task outputs, without jointly checking a final answer and its observations.

Visual Supervision and Process Verification Visual supervision and process verification provide complementary learning signals beyond final-answer correctness (Qin et al., 2025; Xiao et al., 2026; Li et al., 2025; Pronesti et al., 2026). Existing approaches supervise intermediate visual representations or assess a model's observations and reasoning steps. For example, CoVT trains continuous visual tokens to reconstruct features from vision experts (Qin et al., 2025). More recently, PEARL constructs verifiable perception questions for each reasoning task and uses auxiliary perception rollouts to guide subsequent reasoning updates (Zhang et al., 2026b). However, feature reconstruction does not verify the numerical observations stated in a response, and PEARL checks perception through separately generated answers. These objectives do not directly check whether the reported measurements are accurate and support the conclusion within the same reasoning response.

3 METHODS

3.1 OVERVIEW

ASPECT learns to perceive pathology images, report quantitative observations, and reason from them to an answer (Figure 2). Built on Qwen3-VL-8B (Bai et al., 2025), it generates eight pathology feature tokens and six cell tokens before producing the textual observations and answer. We supervise their hidden states through pathology feature reconstruction, cell feature alignment, and count prediction. Three-stage SFT connects these representations to observation and answer generation; RL then rewards correct answers supported by the reported measurements. PathoVernier evaluates this connection through 759 questions spanning four quantitative tasks and more than 15 organs, with reference measurements for every answer.

3.2 PATHOLOGY VISUAL SUPERVISION

Cell composition analysis requires both recognizing cell appearance and measuring abundance. Supervising intermediate visual tokens can retain image information within autoregressive generation (Qin et al., 2025; Li et al., 2026; Yang et al., 2026). For composition analysis, these representations must also encode cell identity and abundance. We therefore pair pathology feature reconstruction with cell feature alignment and count supervision, connecting intermediate visual states to the quantitative observations needed for answering. In the complete response, the two token groups, abbreviated as `<uni>` and `<cell>`, appear in the `<think>` block, followed by observations in `<observe>`, and reasoning with a final answer in `<answer>`. Let $\boldsymbol{H}_{\mathrm{u}} \in \mathbb{R}^{8\times d}$ and $\boldsymbol{H}_{\mathrm{c}} \in \mathbb{R}^{6\times d}$ denote the final-layer hidden states at the two groups, where d is the language model's hidden dimension. The following losses train these states to encode the visual information used in subsequent observation and answer generation.

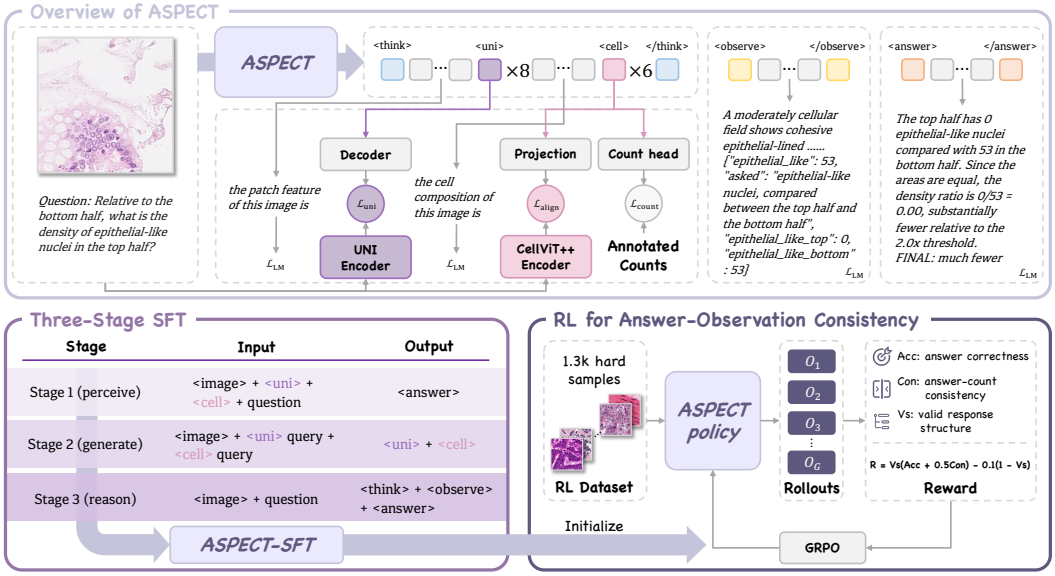


Figure 2: Overview of ASPECT. Pathology feature reconstruction, cell feature alignment, and count supervision train the visual-token representations. Three-stage SFT teaches the model to perceive, generate, and reason with explicit measurements. Subsequent RL rewards answer correctness and consistency with the reported counts.

Pathology feature reconstruction. To preserve pathology appearance in the <uni> states, we reconstruct features $\boldsymbol{F}_{\mathrm{UNI}}(x)$ extracted from image x by a frozen UNI encoder (Chen et al., 2024c). A cross-attention decoder takes learned queries $\boldsymbol{U}$ and the projected token states as inputs:

$$\widehat{\boldsymbol{F}}_{\mathrm{u}} = \mathrm{Dec}_{\mathrm{u}}\big(\boldsymbol{U}, \mathrm{norm}(\boldsymbol{H}_{\mathrm{u}}\boldsymbol{W}_{\mathrm{u}})\big), \quad \mathcal{L}_{\mathrm{uni}} = \Big\|\widehat{\boldsymbol{F}}_{\mathrm{u}} - \boldsymbol{F}_{\mathrm{UNI}}(x)\Big\|_F^2. \tag{1}$$

Here, $\widehat{\boldsymbol{F}}_{\mathrm{u}}$ denotes the reconstructed features, $\boldsymbol{W}_{\mathrm{u}}$ is a learned projection, norm applies L_2 normalization, and $\|\cdot\|_F$ is the Frobenius norm. The cross-attention decoder uses shared learned queries $\boldsymbol{U}$ and the projected states as keys and values, so image-specific information must pass through $\boldsymbol{H}_{\mathrm{u}}$.

Cell feature alignment. With guidance from pathologists, we map source annotations to five target categories: tumor cells, normal epithelial cells, stromal-like cells, lymphocytes, and other inflammatory cells. Finer subtypes within a target category are merged; coarse labels spanning several targets retain their group membership. The first five <cell> tokens represent these categories, and the sixth summarizes all retained, mappable instances. A frozen CellViT++ model (Hörst et al., 2026) supplies instance features $\boldsymbol{u}_j \in \mathbb{R}^{d_T}$, where d_T is the teacher feature dimension. We pool features within annotated masks or predicted instances; NuCLS contributes counts but no alignment targets. For token $s \in \{1, \ldots, 6\}$, let $\mathcal{J}_s$ index its contributing instances. For a non-empty group, its target $\boldsymbol{t}_s$ averages individually normalized features:

$$\overline{\boldsymbol{u}}_j = \frac{\boldsymbol{u}_j}{\|\boldsymbol{u}_j\|_2}, \qquad \boldsymbol{t}_s = \frac{1}{|\mathcal{J}_s|}\sum_{j\in\mathcal{J}_s}\overline{\boldsymbol{u}}_j. \tag{2}$$

An instance with a coarse label contributes to each corresponding category target: an inflammatory instance, for example, supervises both lymphocyte and other-inflammatory features. The sixth token pools each retained instance once; excluded labels are listed in Appendix A.2. We align token states to these targets through

$$\mathcal{L}_{\mathrm{align}} = \frac{1}{d_T|\mathcal{V}|}\sum_{s\in\mathcal{V}}\|\boldsymbol{W}_{\mathrm{c}}(\boldsymbol{h}_s + \boldsymbol{r}_s) - \boldsymbol{t}_s\|_2^2, \tag{3}$$

where $\boldsymbol{h}_s \in \mathbb{R}^d$ is the transposed s-th row of $\boldsymbol{H}_{\mathrm{c}}$, $\boldsymbol{r}_s \in \mathbb{R}^d$ is a learned role embedding, and $\boldsymbol{W}_{\mathrm{c}} \in \mathbb{R}^{d_T\times d}$ is a learned projection. The set $\mathcal{V} = \{s : |\mathcal{J}_s| > 0\}$ excludes empty instance groups.

Count supervision. Mean pooling summarizes appearance without explicitly preserving instance counts. We therefore train the cell-token states to predict the image-wide abundance of the five target categories. An MLP g predicts the five image-wide category counts in log space, $\boldsymbol{\ell} = g(\mathrm{vec}(\boldsymbol{H}_{\mathrm{c}})) \in \mathbb{R}^5$, where vec concatenates all six token states. For aggregation, we recover nonnegative counts as $\hat{n}_k = \max\{0, \exp(\ell_k) - 1\}$. Let $\mathcal{D}$ index categories with individually resolved counts n_k, and let $\mathcal{G}$ contain groups $G \subseteq \{1, \ldots, 5\}$ with annotated combined counts n_G. We optimize

$$\begin{aligned}\mathcal{L}_{\mathrm{count}} = & \sum_{k \in \mathcal{D}} \mathrm{SmoothL1}\big(\ell_k, \log(1 + n_k)\big) \\ & + \sum_{G \in \mathcal{G}} \mathrm{SmoothL1}\left(\log\left(1 + \sum_{k \in G} \hat{n}_k\right), \log(1 + n_G)\right).\end{aligned} \tag{4}$$

Merged fine labels supervise individual target counts, whereas coarse labels supervise their combined abundance. For example, a coarse inflammatory annotation aligns both inflammatory token representations while supervising the sum of lymphocyte and other-inflammatory counts.

The frozen teachers and visual-supervision modules are used only during SFT. The count head supervises image-wide abundance; the language model learns to generate regional measurements through response-text supervision. At inference, it generates visual tokens, observations, and answers autoregressively. Non-cellular training samples use only the `<uni>` group before answer generation.

3.3 Supervised Fine-Tuning

The visual tokens must first acquire useful image representations, then be generated as part of a response and used to support an answer. We organize SFT into three stages, as shown in Figure 2. In *Perceive*, the image, question, and visual tokens are supplied in the user prompt. Visual supervision shapes the token states while the model learns to answer from this context. In *Generate*, the model predicts the visual-token block from the image and a feature query, learning to produce the representations that were previously supplied in the prompt. In *Reason*, it generates the complete response: visual tokens in `<think>`, measurements in `<observe>`, and reasoning with a final answer in `<answer>`. This stage trains the representations, reported measurements, and answer within the same generation sequence.

Cross-entropy covers the entire assistant target and masks the user prompt, supervising visual-token generation in stages two and three. Visual losses apply throughout:

$$\mathcal{L}_{\mathrm{SFT}} = \mathcal{L}_{\mathrm{LM}} + \lambda_{\mathrm{u}}\mathcal{L}_{\mathrm{uni}} + \lambda_{\mathrm{a}}\mathcal{L}_{\mathrm{align}} + \lambda_{\mathrm{c}}\mathcal{L}_{\mathrm{count}}, \tag{5}$$

where $\mathcal{L}_{\mathrm{LM}}$ is assistant-token cross-entropy and the λ coefficients weight the three visual losses. A visual loss is zero when its token group or supervision is unavailable. Training updates LoRA parameters, visual-token parameters, and visual-supervision modules. For quantitative examples, programs derive counts and reference answers from nucleus annotations; a language model supplies descriptions and reasoning, checked against those values. Data, training details, and response templates appear in Appendices A, B, and D.3.

3.4 Reinforcement Learning for Answer–Observation Consistency

SFT provides consistent reference responses, but inference requires reasoning from the model's own estimated counts. An answer-only reward can accept a correct answer that contradicts those counts. We therefore train the model on its generated responses, rewarding both answer correctness and agreement with its observations.

Answer correctness and count consistency. For question u, let Q denote reference counts and $a^\star = f_u(Q)$ the complete reference answer. A response reports counts C and answer $\hat{a}$. Let g_u denote the count-based task check and π_u the corresponding answer projection. We define

$$\begin{aligned}\mathrm{Acc} = \mathbf{1}[\hat{a} = a^\star], \qquad \mathrm{Con} = \mathbf{1}[g_u(C) = \pi_u(\hat{a})], \\ R = V_s\big(\mathrm{Acc} + \beta\,\mathrm{Con}\big) - \gamma(1 - V_s).\end{aligned} \tag{6}$$

Acc checks the complete answer, whereas Con compares the count-derived decision with the corresponding part of the model's own answer. For multi-step composition, Con checks region selection

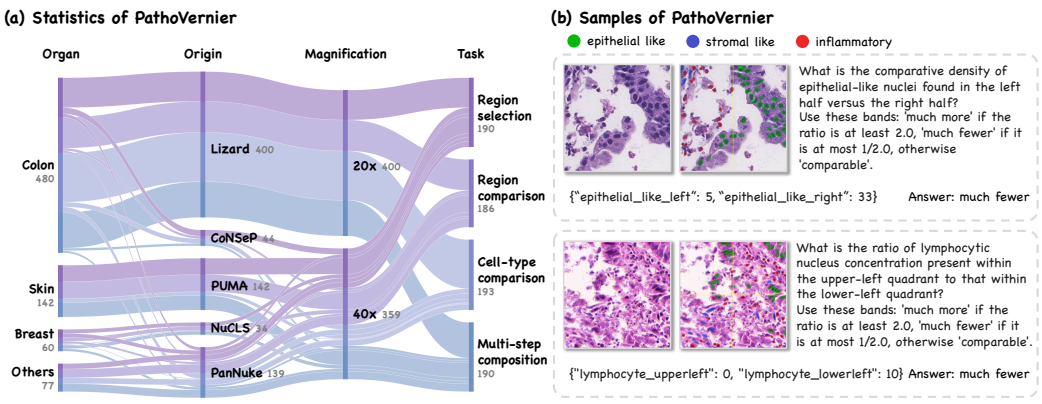


Figure 3: PathoVernier overview. (a) Distribution of 759 questions across organs, sources, magnifications, and tasks; flow widths represent question counts. (b) Example questions with reference counts.

only. We set $\mathrm{Con} = 0$ when either side is undeterminable. The coefficients β and γ weight the consistency bonus and invalid-response penalty. The structural indicator V_s requires parseable observation JSON, nonnegative counts and a parseable final answer. Answer correctness and consistency are evaluated with deterministic rules.

Policy optimization. Starting from ASPECT-SFT, we optimize the policy with GRPO (Shao et al., 2024). We sample a group of responses for each prompt and normalize their rewards within the group to obtain relative advantages for the clipped policy update. LoRA parameters in both the visual and language components are updated. Appendices A.4 and B.3 detail the RL data and optimization settings.

3.5 PathoVernier Benchmark

Data statistics and task coverage. PathoVernier evaluates whether pathology VLMs can answer questions based on accurate cell measurements. As shown in Figure 3(a), it contains 759 expert-reviewed questions across different organs, with images acquired at approximately $20\times$ and $40\times$ magnification. The images originate from five datasets with nucleus-level annotations: Lizard (Graham et al., 2021), PUMA (Schuiveling et al., 2025), PanNuke (Gamper et al., 2020), CoNSeP (Graham et al., 2019), and NuCLS (Amgad et al., 2022). The questions are approximately balanced across four tasks. *Region selection* identifies the region with the highest density of a specified cell type. *Region comparison* compares one cell type across two regions, whereas *cell-type comparison* compares two types within the same image. *Multi-step composition analysis* requires selecting the densest region and then determining a cell proportion category within it. These tasks assess cell recognition, spatial localization, and quantitative reasoning, with reference measurements accompanying every answer.

Question construction and annotation. We filter source images for task suitability and balance while keeping them disjoint from training images. We then map the original nucleus labels to the cell categories used in the questions and compute reference counts from the nucleus annotations. For a region $\mathcal{R}$ and a cell category c, the annotated count is

$$q_{\mathcal{R},c} = \sum_{j \in \mathcal{J}_c} \mathbf{1}[\mathbf{p}_j \in \mathcal{R}], \tag{7}$$

where $\mathcal{J}_c$ indexes retained nuclei belonging to category c, including its constituent classes for aggregate categories, and $\mathbf{p}_j$ is nucleus j's centroid. Four templates convert the required counts Q and rule-derived answers $a^\star = f_u(Q)$ into 786 candidate questions. GPT-5.6-sol (OpenAI, 2026b) rewrites their wording while preserving numerical rules and answers. Experts review each question against its image and annotation overlay, retaining 759 valid items (Figure 3(b)). Appendices A.5 and D.2 detail filtering, rewriting, and expert review.

Table 1: Comparison on PathoVernier and three external pathology benchmarks. ASPECT leads on all PathoVernier metrics and improves over Qwen3-VL-8B on all external benchmarks. **Bold** and underlined values denote the best and second-best results, respectively. Dashes indicate CA coverage below 30%; Table C.11 reports coverage and unconditional Count Acc.

Models	PathoVernier			PathCLS	PathVQA	Quilt-VQA
	Acc↑	CA↑	RAWR↓	Acc↑	Acc↑	Acc↑
Closed-source Models						
GPT-5.5 (OpenAI, 2026a)	0.59	0.65	0.76	0.55	0.69	**0.69**
Gemini-3.1-Pro (Google DeepMind, 2026)	0.63	0.71	0.68	**0.67**	**0.78**	0.65
Open-source Models						
Qwen3-VL-8B (Bai et al., 2025)	0.37	0.53	0.85	0.37	0.67	0.58
Qwen3-VL-32B (Bai et al., 2025)	0.44	0.50	0.82	0.45	0.65	0.62
InternVL3-8B (Zhu et al., 2025)	0.30	0.38	0.86	0.39	0.69	0.64
InternVL3-38B (Zhu et al., 2025)	0.45	0.45	0.83	0.41	0.71	0.64
Medical and Pathology-tuned Models						
Quilt-LLaVA-7B (Seyfioglu et al., 2024)	0.12	-	-	0.24	0.54	0.64
PathGen-LLaVA-13B (Sun et al., 2025)	0.12	-	-	0.52	0.62	0.66
HuatuoGPT-Vision-7B (Chen et al., 2024a)	0.30	-	-	0.34	0.64	0.55
Patho-R1-7B (Zhang et al., 2026c)	0.22	-	-	0.44	0.56	0.68
ASPECT-8B(ours)	**0.75**	**0.81**	**0.49**	0.54	0.74	0.67
Δ (*vs base model Qwen3-VL-8B*)	**+0.38**	**+0.28**	**-0.36**	**+0.17**	**+0.07**	**+0.09**

Evaluation metrics. We report answer accuracy $\mathrm{Acc} = \mathbb{E}[\mathbf{1}[\hat{a} = a^\star]]$ and two measures of the reported observations. All expectations denote empirical averages over eligible responses. Let $\mathcal{K}$ index a question's required counts, q_i be its annotated count, and $\mathrm{dom}(C)$ index reported counts C_i. The fraction of required counts within tolerance is

$$S(C, Q) = \frac{1}{|\mathcal{K}|} \sum_{i \in \mathcal{K} \cap \mathrm{dom}(C)} \mathbf{1}\big[|C_i - q_i| \le \tau_i\big], \qquad \tau_i = \max(1, 0.1 q_i). \tag{8}$$

Each count contributes equally; missing counts contribute zero. We adapt coherent accuracy (CA) from VPRM (Pronesti et al., 2026) to compare a count-derived task decision with its reference:

$$\mathrm{CA} = \mathbb{E}[\mathbf{1}[g_u(C) = \pi_u(a^\star)] \,|\, g_u(C) \text{ is defined}]\,. \tag{9}$$

Here, g_u and π_u denote the task check and answer projection defined in Appendix C.2. Additionally, right answer, wrong reason (RAWR) measures the fraction of required counts outside tolerance among correct responses with complete measurements:

$$\mathrm{RAWR} = 1 - \mathbb{E}[S(C, Q) \,|\, \hat{a} = a^\star,\ \mathcal{K} \subseteq \mathrm{dom}(C)]\,. \tag{10}$$

Lower RAWR indicates fewer out-of-tolerance counts. Appendix C.2 specifies eligibility for both conditional metrics.

4 Experiments

Performance on PathoVernier. Table 1 shows that ASPECT leads on all three PathoVernier metrics, with relative gains of approximately 19.2% in Acc and 12.9% in CA, and a 28.1% reduction in RAWR over Gemini-3.1-Pro, the strongest baseline. ASPECT thus improves both task decisions and the quantitative observations underlying them. The reduction in RAWR directly addresses our motivating observation: even among correct responses, ASPECT's reported counts more often match the annotated measurements. In contrast, all medical- and pathology-tuned baselines fall below Qwen3-VL-8B in PathoVernier accuracy and have CA coverage below 30%, so we omit their CA and RAWR. Notably, PathGen-LLaVA and Patho-R1 outperform Qwen3-VL-8B on PathCLS yet underperform it on PathoVernier. This ranking reversal shows that conventional pathology question-answering performance does not guarantee reliable quantitative image understanding. Appendices E.1–E.4 report task- and source-level results, confidence intervals, and tolerance sensitivity.

Performance on external pathology benchmarks. ASPECT also improves over its Qwen3-VL-8B backbone on all three external pathology benchmarks, with relative accuracy gains of approximately 45.9% on PathCLS (Sun et al., 2024), 10.4% on PathVQA (He et al., 2020), and 15.5% on

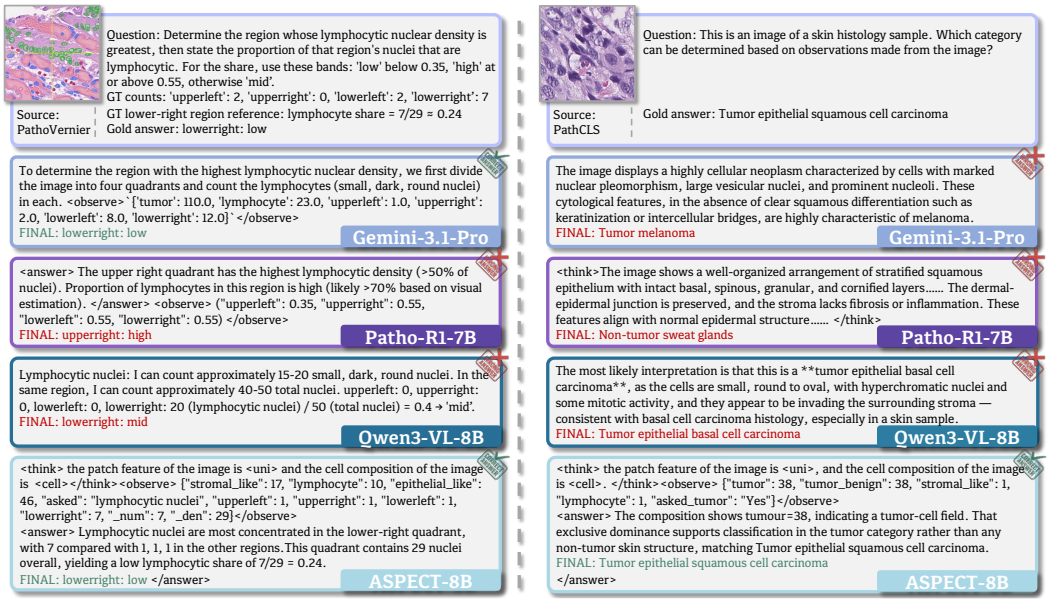


Figure 4: Qualitative comparison on PathoVernier (left) and PathCLS (right). Responses are condensed and reformatted to highlight key observations and predictions. ASPECT reports more accurate regional counts on PathoVernier and predicts the reference class on PathCLS.

Table 2: Ablations of visual supervision and the SFT curriculum on PathoVernier. All variants are evaluated before RL with matched total update budgets.

Configuration	Acc↑	CA↑	RAWR↓	Count Acc↑
ASPECT-SFT	0.70	0.80	0.53	0.44
w/o visual supervision	0.64	0.77	0.59	0.32
w/o pathology feature reconstruction	0.68	0.75	0.56	0.27
w/o cell feature alignment	0.67	0.77	0.54	0.29
w/o count supervision	0.72	0.78	0.57	0.36
Direct Reason training (Stage 3)	0.62	0.74	0.56	0.31

the pathology-image subset of Quilt-VQA (Seyfioglu et al., 2024). The gains on classification and question answering show that training for cellular composition also benefits pathology tasks that do not explicitly request measurements.

Qualitative comparison. Figure 4 summarizes predictions from four models. On PathoVernier, Gemini and ASPECT give the same correct answer, but Gemini overestimates the lower-right lymphocyte count (12 versus 7). ASPECT's regional counts fall within tolerance, and its reported proportion, $7/29 \approx 0.24$, correctly supports the `low` category. Qwen3-VL-8B selects the correct region but misjudges the proportion; Patho-R1 errs on both steps. This contrast shows how answer accuracy can conceal measurement errors. On PathCLS, ASPECT predicts the reference class while all three baselines misclassify the image, illustrating a classification example beyond composition analysis. Full responses for these and additional cases are provided in Appendix F.

Ablation on visual supervision and training curriculum. Table 2 ablates visual supervision and the SFT curriculum on PathoVernier. Count Acc averages $S(C, Q)$ over all questions, while RAWR measures count errors only among correct answers with complete measurements. ASPECT-SFT achieves 0.44 Count Acc, compared with 0.296 for Gemini-3.1-Pro and 0.068 for Qwen3-VL-8B. Removing pathology feature reconstruction or cell alignment reduces Count Acc to 0.27 or 0.29. Removing count supervision slightly improves Acc but lowers Count Acc to 0.36 and increases RAWR to 0.57. Thus, reported measurements can support the correct conclusion while

Table 3: RL reward ablations on PathoVernier. The listed rewards are the internal term r; all RL variants retain the structural validity gate and the invalid-response penalty.

Configuration	RL Reward	Acc↑	CA↑	RAWR↓	Count Acc↑	Consistency↑
ASPECT-SFT	N/A	0.70	0.80	0.53	0.44	0.94
Answer only	Acc	0.72	0.79	0.55	0.43	0.92
Consistency only	0.5 Con	0.69	0.76	0.51	0.41	0.98
Answer + count score	Acc + 0.5 S	0.74	0.80	0.50	0.45	0.96
ASPECT-8B	Acc + 0.5 Con	0.75	0.81	0.49	0.47	0.99

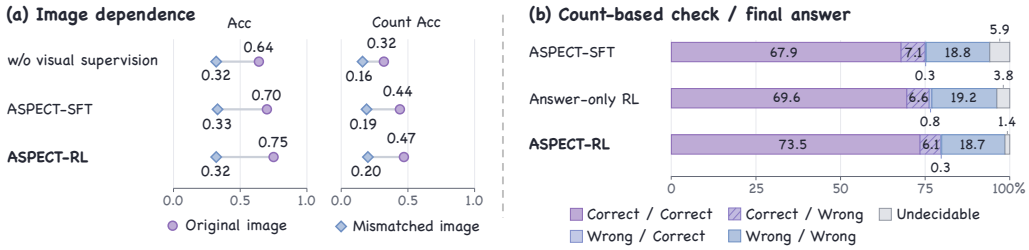


Figure 5: Image dependence and task correctness on PathoVernier. (a) Performance with original and mismatched images. (b) Joint outcomes of the count-based check and final-answer correctness.

remaining inaccurate, reinforcing the need to evaluate observations alongside answers. The staged SFT configuration also improves answer and count accuracy over QA-only Direct Reason training.

Ablation on reinforcement learning rewards. Table 3 compares reward variants from the same SFT checkpoint, retaining $R = V_s r - 0.1(1 - V_s)$ and varying only r, with count score $S(C, Q)$. Consistency averages Con over responses with determinable count-derived decisions and answer components, measuring agreement with the model's own decision; CA instead uses the reference decision. Answer-only RL improves Acc while degrading count accuracy and consistency. Conversely, consistency-only RL reaches 0.98 agreement without improving answer or count accuracy over SFT. ASPECT combines these complementary signals and achieves the highest Acc and Count Acc and the lowest RAWR among the tested rewards.

Image dependence and task correctness. Figure 5(a) shows that mismatching images largely removes the Acc gains and reduces the Count Acc gains, indicating that ASPECT's improvements depend on relevant visual evidence. Figure 5(b) further examines a gold-anchored count-based check alongside final-answer correctness. ASPECT-RL passes both checks on 73.5% of questions, versus 67.9% for SFT and 69.6% for answer-only RL. For multi-step composition, the count-based check assesses the reference region, whereas final-answer accuracy requires both the region and proportion band (Appendix C.3).

5 Conclusion

We presented ASPECT and PathoVernier to connect fine-grained visual perception with pathology reasoning. ASPECT supervises intermediate visual tokens for pathology appearance and cell abundance, then learns to generate and use quantitative observations through staged SFT and RL. PathoVernier evaluates the resulting answers together with their supporting measurements. ASPECT improves both answer accuracy and count fidelity, with further gains over its backbone on three external pathology benchmarks. Ablations and image-mismatch experiments show the value of visual supervision and the dependence of these gains on relevant images. Together, these results demonstrate the value of treating quantitative observations as explicit targets for pathology VLM training and evaluation.

AI USE STATEMENT

In this work, we used generative AI tools to generate synthetic datasets, implement methods, assist with translation, clean and reformat datasets, and support qualitative and thematic data analysis. The generated data consisted of textual descriptions and reasoning for training, and language models also assisted with rewriting questions. These tools did not generate pathology images or ground-truth nucleus annotations. We have not used generative AI tools to design or provide feedback on research methodology and experiments, develop theoretical models or conceptual frameworks, propose or refine hypotheses, or interpret results. Formulating mathematical claims, providing critical ingredients for proving mathematical claims, and writing proofs are not applicable to this work. Additionally, we used generative AI tools to suggest experimental parameters, summarize and analyze existing literature, and search for information. We have reviewed all AI-assisted work. The authors reviewed AI-assisted code and data-processing workflows. Training text underwent automatic checks against reference counts and answers, while experts reviewed benchmark questions against source images and annotations. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The supplementary archive contains core SFT and RL training code, PathoVernier inference and scoring code, benchmark questions with reference counts and patch indices, and saved ASPECT outputs for result verification. Appendix B specifies the model configuration, training hyperparameters, and algorithms, while Appendix C details the evaluation protocols and metric definitions. The prompts are provided in Appendix D, and the additional analyses include confidence intervals and count-tolerance sensitivity in Appendices E.3 and E.4. The ASPECT model weights will be released publicly. Source images must be obtained from the original providers, as described in Appendix A.1.

REFERENCES

Mohamed Amgad, Lamees A Atteya, Hagar Hussein, Kareem Hosny Mohammed, Ehab Hafiz, Maha AT Elsebaie, Ahmed M Alhusseiny, Mohamed Atef AlMoslemany, Abdelmagid M Elmatboly, Philip A Pappalardo, et al. Nucls: A scalable crowdsourcing approach and dataset for nucleus classification and segmentation in breast cancer. *GigaScience*, 11:giac037, 2022.

Mohammad Asadi, Jack W O'Sullivan, Fang Cao, Tahoura Nedaee, Kamyar Rajabalifardi, Fei-Fei Li, Ehsan Adeli, and Euan Ashley. Mirage: The illusion of visual understanding. *arXiv preprint arXiv:2603.21687*, 2026.

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

Junying Chen, Chi Gui, Ruyi Ouyang, Anningzhe Gao, Shunian Chen, Guiming Hardy Chen, Xidong Wang, Zhenyang Cai, Ke Ji, Xiang Wan, et al. Towards injecting medical visual knowledge into multimodal llms at scale. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 7346–7370, 2024a.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating large vision-language models? *Advances in Neural Information Processing Systems*, 37:27056–27087, 2024b.

Richard J Chen, Tong Ding, Ming Y Lu, Drew FK Williamson, Guillaume Jaume, Andrew H Song, Bowen Chen, Andrew Zhang, Daniel Shao, Muhammad Shaban, et al. Towards a general-purpose foundation model for computational pathology. *Nature Medicine*, 30(3):850–862, 2024c.

Zongyi Chen, Yu Liang, Jie Lin, and Liansheng Wang. Pathview-bench: Can multimodal large language models achieve fine-grained multiscale understanding of pathology images? *arXiv preprint arXiv:2607.28318*, 2026.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A Smith, Wei-Chiu Ma, and Ranjay Krishna. Blink: Multimodal large language models can see

but not perceive. In *Proceedings of the European Conference on Computer Vision*, pp. 148–166. Springer, 2024.

Jevgenij Gamper, Navid Alemi Koohbanani, Ksenija Benes, Simon Graham, Mostafa Jahanifar, Syed Ali Khurram, Ayesha Azam, Katherine Hewitt, and Nasir Rajpoot. Pannuke dataset extension, insights and baselines. *arXiv preprint arXiv:2003.10778*, 2020.

Google DeepMind. Gemini 3.1 pro. Model Card, February 2026. URL https://deepmind.google/models/model-cards/gemini-3-1-pro/. Published February 19, 2026.

Simon Graham, Quoc Dang Vu, Shan E Ahmed Raza, Ayesha Azam, Yee Wah Tsang, Jin Tae Kwak, and Nasir Rajpoot. Hover-net: Simultaneous segmentation and classification of nuclei in multi-tissue histology images. *Medical Image Analysis*, 58:101563, 2019.

Simon Graham, Mostafa Jahanifar, Ayesha Azam, Mohammed Nimir, Yee-Wah Tsang, Katherine Dodd, Emily Hero, Harvir Sahota, Atisha Tank, Ksenija Benes, et al. Lizard: A large-scale dataset for colonic nuclear instance segmentation and classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 684–693, 2021.

Xuehai He, Yichen Zhang, Luntian Mou, Eric Xing, and Pengtao Xie. Pathvqa: 30000+ questions for medical visual question answering. *arXiv preprint arXiv:2003.10286*, 2020.

Yunqi Hong, Kuei-Chun Kao, Liam Edwards, Nein-Tzu Liu, Chung-Yen Huang, Alex Oliveira-Kowaleski, Cho-Jui Hsieh, and Neil Y. C. Lin. Adaptive diagnostic reasoning framework for pathology with multimodal large language models. *Communications Medicine*, 6:236, 2026.

Fabian Hörst, Moritz Rempe, Helmut Becker, Lukas Heine, Julius Keyl, and Jens Kleesiek. Cellvit++: Energy-efficient and adaptive cell segmentation and classification using foundation models. *Computer Methods and Programs in Biomedicine*, pp. 109206, 2026.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Shengyi Hua, Jianfeng Wu, Tianle Shen, Kangzhe Hu, Zhongzhen Huang, Shujuan Ni, Zhihong Zhang, Yuan Li, Zhe Wang, and Xiaofan Zhang. PathFound: An agentic multimodal model activating evidence-seeking pathological diagnosis. *Medical Image Analysis*, 113:104200, 2026.

Songhan Jiang, Fengchun Liu, Ziyue Wang, Linghan Cai, and Yongbing Zhang. Pathreasoner-r1: Instilling structured reasoning into pathology vision-language model via knowledge-guided policy optimization. *arXiv preprint arXiv:2601.21617*, 2026.

Jakob Nikolas Kather, Niels Halama, and Alexander Marx. 100,000 histological images of human colorectal cancer and healthy tissue. Zenodo, 2018. URL https://doi.org/10.5281/zenodo.1214456.

Bangzheng Li, Ximeng Sun, Jiang Liu, Ze Wang, Jialian Wu, Xiaodong Yu, Emad Barsoum, Muhao Chen, and Zicheng Liu. Latent visual reasoning. In *Proceedings of the International Conference on Learning Representations*, volume 2026, pp. 148076–148090, 2026.

Zongxia Li, Wenhao Yu, Chengsong Huang, Zhenwen Liang, Rui Liu, Fuxiao Liu, Jingxi Che, Dian Yu, Jordan Boyd-Graber, Haitao Mi, et al. Self-rewarding vision-language model via reasoning decomposition. *arXiv preprint arXiv:2508.19652*, 2025.

Neal I Lindeman, Philip T Cagle, Dara L Aisner, Maria E Arcila, Mary Beth Beasley, Eric H Bernicker, Carol Colasacco, Sanja Dacic, Fred R Hirsch, Keith Kerr, et al. Updated molecular testing guideline for the selection of lung cancer patients for treatment with targeted tyrosine kinase inhibitors: guideline from the college of american pathologists, the international association for the study of lung cancer, and the association for molecular pathology. *Archives of Pathology & Laboratory Medicine*, 142(3):321–346, 2018.

Ming Y Lu, Bowen Chen, Drew FK Williamson, Richard J Chen, Melissa Zhao, Aaron K Chow, Kenji Ikemura, Ahrong Kim, Dimitra Pouli, Ankush Patel, et al. A multimodal generative ai copilot for human pathology. *Nature*, 634(8033):466–473, 2024.

OpenAI. Gpt-5.5 system card. April 2026a. URL https://deploymentsafety.openai.com/gpt-5-5/gpt-5-5.pdf.

OpenAI. GPT-5.6 Sol. https://developers.openai.com/api/docs/models/gpt-5.6-sol, 2026b. OpenAI model documentation. Accessed: 2026-09-23.

Massimiliano Pronesti, Anya Belz, and Yufang Hou. Beyond outcome verification: Verifiable process reward models for structured reasoning. In *Findings of the Association for Computational Linguistics*, pp. 32187–32202, 2026.

Yiming Qin, Bomin Wei, Jiaxin Ge, Konstantinos Kallidromitis, Stephanie Fu, Trevor Darrell, and XuDong Wang. Chain-of-visual-thought: Teaching vlms to see and think better with continuous visual tokens. *arXiv preprint arXiv:2511.19418*, 2025.

Pooyan Rahmanzadehgervi, Logan Bolton, Mohammad Reza Taesiri, and Anh Totti Nguyen. Vision language models are blind. In *Proceedings of the Asian Conference on Computer Vision*, pp. 293–309. Springer, 2024.

Roberto Salgado, Carsten Denkert, Sandra Demaria, Nicolas Sirtaine, Frederick Klauschen, Giancarlo Pruneri, Stephan Wienert, Gert Van den Eynden, Frederick L Baehner, Frederique Pénault-Llorca, et al. The evaluation of tumor-infiltrating lymphocytes (tils) in breast cancer: recommendations by an international tils working group 2014. *Annals of Oncology*, 26(2):259–271, 2015.

Mark Schuiveling, Hong Liu, Daniel Eek, Gerben E Breimer, Karijn PM Suijkerbuijk, Willeke AM Blokx, and Mitko Veta. A novel dataset for nuclei and tissue segmentation in melanoma with baseline nuclei segmentation and tissue segmentation benchmarks. *GigaScience*, 14:giaf011, 2025.

Mehmet Saygin Seyfioglu, Wisdom O Ikezogwo, Fatemeh Ghezloo, Ranjay Krishna, and Linda Shapiro. Quilt-llava: Visual instruction tuning by extracting localized narratives from open-source histopathology videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13183–13192. IEEE, 2024.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Yuxuan Sun, Hao Wu, Chenglu Zhu, Sunyi Zheng, Qizi Chen, Kai Zhang, Yunlong Zhang, Dan Wan, Xiaoxiao Lan, Mengyue Zheng, et al. Pathmmu: A massive multimodal expert-level benchmark for understanding and reasoning in pathology. In *Proceedings of the European Conference on Computer Vision*, pp. 56–73. Springer, 2024.

Yuxuan Sun, Yunlong Zhang, Yixuan Si, Chenglu Zhu, Kai Zhang, Zhongyi Shui, Jingxiong Li, Xuan Gong, Xinheng Lyu, Tao Lin, et al. Pathgen-1.6 m: 1.6 million pathology image-text pairs generation through multi-agent collaboration. In *Proceedings of the International Conference on Learning Representations*, volume 2025, pp. 94611–94653, 2025.

Tong Xiao, Xin Xu, Zhenya Huang, Hongyu Gao, Quan Liu, Qi Liu, and Enhong Chen. Perception-r1: Advancing multimodal reasoning capabilities of mllms via visual perception reward. In *Proceedings of the International Conference on Learning Representations*, volume 2026, pp. 26868–26898, 2026.

Zeyuan Yang, Xueyang Yu, Delin Chen, Maohao Shen, and Chuang Gan. Machine mental imagery: Empower multimodal reasoning with latent visual tokens. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 33510–33520, 2026.

Chengyang Zhang, Wenchuan Zhang, Bo Li, Xinyu Liu, Jiaming Yang, Mengran Li, Chenxun Deng, Jie Chen, Yang Zhang, Wei Ju, et al. Do pathology vision-language models truly see pathology? *arXiv preprint arXiv:2607.21065*, 2026a.

Chi Zhang, Haibo Qiu, Qiming Zhang, Yufei Xu, Zhixiong Zeng, Siqi Yang, Peng Shi, Lin Ma, and Jing Zhang. Perceptual-evidence anchored reinforced learning for multimodal reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 41111–41120, 2026b.

Wenchuan Zhang, Penghao Zhang, Jingru Guo, Tao Cheng, Jie Chen, Shuwan Zhang, Zhang Zhang, Yuhao Yi, and Hong Bu. Patho-r1: A multimodal reinforcement learning-based pathology expert reasoner. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 28418–28426, 2026c.

Yikang Zhou, Tao Zhang, Shilin Xu, Shihao Chen, Qianyu Zhou, Yunhai Tong, Shunping Ji, Jiangning Zhang, Lu Qi, and Xiangtai Li. Are they the same? exploring visual correspondence shortcomings of multimodal llms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17663–17674. IEEE, 2025.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

APPENDIX

CONTENTS OF THE APPENDIX

- A. Data Sources and Construction
- B. Implementation Details and Algorithms
- C. Evaluation Protocols and Metric Definitions
- D. Prompts and Response Formats
- E. Additional Quantitative Results
- F. Complete Case Studies and Failure Analysis
- G. Limitations and Future Work

A DATA SOURCES AND CONSTRUCTION

A.1 SOURCE DATASETS AND DATA SPLITS

We use seven public datasets for training and benchmark construction (Table A.1). Lizard (Graham et al., 2021), PUMA (Schuiveling et al., 2025), PanNuke (Gamper et al., 2020), CoNSeP (Graham et al., 2019), and NuCLS (Amgad et al., 2022) provide nucleus-level annotations for quantitative supervision and PathoVernier construction. The first four provide instance masks and cell labels, whereas NuCLS provides nucleus locations and class annotations for the subset used in this work. NCT-CRC-HE-100K (Kather et al., 2018) and the H&E subset of the PathVQA (He et al., 2020) training split add tissue-classification and pathology question-answering examples during SFT. For NCT-CRC-HE-100K, cellular supervision combines detected nuclei with patch-level tissue labels. For PathVQA, it uses predicted nucleus locations and native cell-type predictions. Neither source provides reference nucleus-level annotations.

Data are partitioned according to the available source identifiers: source images for Lizard, source WSIs and their corresponding ROIs for PUMA, individual patches for PanNuke, and TCGA patient–slide identifiers for NuCLS. CoNSeP follows its official training and test partition. NCT-CRC-HE-100K is sampled from its official training set with balanced tissue classes, while PathVQA uses only its official training split after H&E filtering. SFT, RL, validation, and PathoVernier use mutually disjoint image sets. The SFT collection contains 19,422 visual-supervision examples and 13,095 question-answer pairs. RL uses 1,372 questions from 1,180 unique images, with 93 validation questions used for checkpoint selection. PathoVernier contains 759 questions from 553 unique patches.

Data access and release. The PathoVernier release will include questions, reference counts, split indices, and scripts for reconstructing the benchmark from the original datasets. Images will be obtained directly from the respective providers rather than redistributed with the benchmark. The release documentation will retain source attribution and specify the applicable licenses and access terms for source-derived materials.

A.2 CELL CATEGORY HARMONIZATION

With guidance from pathologists, we harmonize source labels into five categories: tumor cells, normal epithelial cells, stromal-like cells, lymphocytes, and other inflammatory cells (Table A.2). Fine-grained labels within a category are merged. Coarse labels retain their aggregate meaning: Lizard epithelial annotations remain epithelial-like, while PanNuke and CoNSeP inflammatory annotations remain inflammatory. For feature alignment, these annotations contribute to both corresponding category targets; count supervision applies only to their combined count.

Background and excluded labels are removed before constructing supervision: Dead in PanNuke, Other/Miscellaneous in CoNSeP, apoptotic nuclei in PUMA, and unlabeled nuclei or apoptotic bodies in NuCLS. These instances contribute neither to category counts nor to the total count or the sixth cell-token target. The sixth token summarizes all retained, mappable instances, each included once.

Table A.1: Data sources and sample counts. SFT examples are separated into visual-supervision and question-answer examples. RL, validation, and PathoVernier columns report question counts. Multiple examples may share an image.

Source	SFT		RL	Validation	PathoVernier
	Visual	VQA			
Lizard (Graham et al., 2021)	2,520	2,944	559	40	400
PUMA (Schuiveling et al., 2025)	2,745	1,010	219	5	142
PanNuke (Gamper et al., 2020)	6,780	2,565	567	20	139
CoNSeP (Graham et al., 2019)	196	133	27	2	44
NuCLS (Amgad et al., 2022)	854	77	0	26	34
NCT-CRC-HE-100K (Kather et al., 2018)	5,938	5,386	0	0	0
PathVQA (He et al., 2020)	389	980	0	0	0
Total	19,422	13,095	1,372	93	759

For sources without nucleus-level annotations, we use two supervision paths (Table A.3). In NCT-CRC-HE-100K, CellViT-SAM-H detects nuclei, and all detected nuclei in a patch inherit the category mapped from its tissue label. Adenocarcinoma epithelium maps to tumor, normal colon mucosa to normal epithelial, stroma and smooth muscle to stromal-like, and lymphocytes to lymphocyte. These assignments provide weak cell-type labels. Adipose, debris, background, and mucus patches receive pathology feature reconstruction without cell-specific supervision. In PathVQA, native CellViT PanNuke predictions follow the PanNuke mapping in Table A.2, including aggregate inflammatory supervision and exclusion of Dead and background predictions.

Benchmark questions use tumor, stromal-like, lymphocyte, epithelial-like, and inflammatory counts. Epithelial-like combines tumor and normal epithelial cells; inflammatory combines lymphocytes and other inflammatory cells. These quantities are obtained by summing resolved categories or retaining the corresponding aggregate annotation. Questions respect source-label granularity: Lizard does not support tumor-specific questions, while PanNuke and CoNSeP do not support lymphocyte-specific questions. Normal epithelial and other inflammatory counts are not queried individually. Cell-type comparisons exclude category pairs with a subset relationship, such as tumor versus epithelial-like.

A.3 SFT Data and Teacher Targets

The SFT collection contains 19,422 visual-supervision records and 13,095 question-answer examples. These records are rendered into stage-specific input–target pairs: the visual-supervision records are used in Generate, while QA records are used in all three stages. The question-answer examples comprise 6,729 programmatically constructed quantitative questions from the five nucleus-annotated datasets, 5,386 classification and related questions from NCT-CRC-HE-100K, and 980 closed-ended yes/no questions from the H&E subset of PathVQA. GPT-5.6-sol generates descriptive prose and reasoning from the image and source-specific reference information. In a subsequent text-only pass, the same model rewrites the `<answer>` reasoning notes while preserving their numerical values and final answers. PathVQA observations are qualitative and contain no count JSON. Automatic checks enforce agreement between the generated prose, reference counts, and answers. Failed generations are retried or discarded; rewrites that alter numerical values or final answers revert to templates.

Visual teacher targets are extracted offline and remain fixed during training. UNI provides image features for pathology feature reconstruction. For Lizard, PUMA, PanNuke, and CoNSeP, CellViT-SAM-H features are mean-pooled within ground-truth nucleus masks to obtain instance embeddings, then normalized and aggregated into six cell-token targets as described in Section 3.2. NuCLS supplies UNI features and category counts derived from point annotations, but does not contribute to cell feature alignment. NCT-CRC-HE-100K and PathVQA use the detected instances and category assignments described in Appendix A.2. PathVQA patches with fewer than three retained detections receive UNI supervision only.

Table A.2: Source-label mappings for cellular supervision. Starred entries retain aggregate labels: feature alignment uses both listed targets, whereas count supervision uses their sum. NuCLS uses these mappings for count supervision only.

Source	Original label(s)	Target category/categories
Lizard	Epithelial	Tumor; normal epithelial*
	Connective tissue	Stromal-like
	Lymphocyte	Lymphocyte
	Plasma, neutrophil, eosinophil	Other inflammatory
PUMA	Tumor	Tumor
	Epithelium	Normal epithelial
	Stroma, endothelium	Stromal-like
	Lymphocyte	Lymphocyte
	Plasma cell, neutrophil, histiocyte, melanophage	Other inflammatory
PanNuke	Neoplastic	Tumor
	Epithelial	Normal epithelial
	Connective/soft tissue	Stromal-like
	Inflammatory	Lymphocyte; other inflammatory*
CoNSeP	Dysplastic/malignant epithelial	Tumor
	Healthy epithelial	Normal epithelial
	Fibroblast, muscle, endothelial	Stromal-like
	Inflammatory	Lymphocyte; other inflammatory*
NuCLS	Tumor, mitotic figure	Tumor
	Ductal epithelium, myoepithelium	Normal epithelial
	Fibroblast, vascular endothelium	Stromal-like
	Lymphocyte	Lymphocyte
	Plasma cell, macrophage, neutrophil, eosinophil	Other inflammatory

Table A.3: Cellular supervision for sources without nucleus-level reference annotations. CRC cell types inherit patch-level tissue labels; PathVQA cell types come from native nucleus predictions.

Source	Label source	Feature-alignment targets	Count supervision
NCT-CRC-HE-100K	Patch-level tissue labels	Mapped patch category for all detected nuclei	Detected nucleus count assigned to that category
PathVQA	Native CellViT PanNuke predictions	Mapped nucleus categories; shared inflammatory targets	Predicted category counts; aggregate inflammatory count

A.4 RL Data Construction

The RL collection contains 1,372 questions from 1,180 unique images drawn from Lizard, PUMA, PanNuke, and CoNSeP. Reference counts are derived from nucleus annotations, and task rules determine the reference answers. RL images are disjoint from those used for SFT and PathoVernier. The collection covers eight skills, including regional selection, composition analysis, and additional counting and comparison tasks (Table A.4).

We select questions by skill category using the SFT model's rollout accuracy to identify skills with greater room for improvement. We retain the available questions for multi-step composition, region selection, dominant cell type, count band prediction, and count comparison, while limiting the more saturated region-comparison, cell-type-comparison, and yes/no categories. This allocation concentrates training on skills where SFT remains less reliable while preserving task diversity. A 93-question validation set is used for checkpoint selection. Reward and optimization settings are provided in Appendix B.3.

Table A.4: Skill composition of the RL training collection. Question selection is based on skill-level SFT performance.

Skill	Questions	Skill	Questions
Multi-step composition	388	Count comparison	85
Region selection	407	Region comparison	60
Dominant cell type	208	Cell-type comparison	60
Count band prediction	124	Yes/no	40
Total			1,372

A.5 PathoVernier Construction and Expert Review

We construct PathoVernier from eligible H&E patches in the five nucleus-annotated datasets. After the category filtering described in Appendix A.2, patches containing fewer than five or at least 300 retained nuclei are excluded. NuCLS contributes only questions whose reference quantities can be computed from point annotations. Candidate questions are sampled with quotas stratified by task, dataset, and answer category to balance task coverage and answer distributions. Four task templates instantiate questions from annotation-derived counts and deterministic decision rules, yielding 786 candidates.

GPT-5.6-sol paraphrases the questions while numerical thresholds and answer options remain fixed. Automatic checks preserve the queried cell types, regions, and relationships; a separate language model checks semantic equivalence against the reference question. Paraphrases that fail these checks are regenerated, with the original template retained when retries are exhausted. An independent implementation recomputes regional assignments and rule-based answers from the annotations and flags discrepancies for manual review. Detailed task rules and rewriting prompts are provided in Appendices C.2 and D.2, respectively.

All 786 candidates are presented through a structured review interface to four board-certified pathologists, with automatically flagged cases prioritized for inspection. Review assesses whether the question matches the image, the reference counts agree with the annotations and visible morphology, and the answer is reliably distinguishable. Particular attention is given to near-tied regional counts and proportions close to category thresholds, especially when small denominators make the answer sensitive to counting discrepancies. Following review, 27 questions are excluded, leaving 759 questions from 553 unique patches with approximately balanced coverage of the four tasks.

A.6 Detailed Dataset Statistics

PathoVernier contains 759 questions from 553 unique patches. The four tasks each account for 24.5–25.4% of the questions, with their source composition detailed in Table A.5. We harmonize the 19 source organ labels into 18 categories by merging Colon and Colorectal (Table A.6). Colon, skin, and breast contribute 480, 142, and 60 questions, respectively. The remaining 15 categories contribute 77 questions and are supplied by PanNuke. All distributions in this section count questions, including those sharing the same patch.

Table A.5: PathoVernier question counts by source dataset and task.

Source	Region selection	Region comparison	Cell-type comparison	Multi-step composition	Total
Lizard	82	85	135	98	400
PUMA	45	47	10	40	142
PanNuke	35	36	33	35	139
CoNSeP	17	7	13	7	44
NuCLS	11	11	2	10	34
Total	190	186	193	190	759

Each patch contains 256×256 pixels at its source resolution. Physical resolution ranges from 0.20 to 0.50 μm/pixel, corresponding to patch widths of 51.2–128 μm (Table A.7). The 400 Lizard questions

use images at approximately $20\times$, while the remaining 359 questions use images at approximately $40\times$. These magnification groups therefore also differ in data source and organ composition.

Table A.6: PathoVernier question counts across 18 harmonized organ categories. Colon includes the Colorectal label from CoNSeP.

Organ	Questions	Organ	Questions
Colon	480	Kidney	4
Skin	142	Ovary	3
Breast	60	Liver	2
Bile duct	15	Pancreas	2
Head and neck	11	Prostate	2
Uterus	10	Stomach	2
Esophagus	9	Testis	2
Cervix	6	Thyroid	2
Lung	6	Bladder	1
Total			759

Table A.7: Image scale by source. Patch width is calculated from the 256×256-pixel inputs and source resolution. Magnifications are approximate objective equivalents.

Source	Resolution (μm/pixel)	Patch width (μm)	Approx. magnification	Questions
Lizard	0.50	128.0	$20\times$	400
PanNuke, CoNSeP	0.25	64.0	$40\times$	183
PUMA	0.22	56.3	$40\times$	142
NuCLS	0.20	51.2	$40\times$	34
Total				759

B Implementation Details and Algorithms

B.1 Model and Training Configurations

ASPECT uses Qwen3-VL-8B (Bai et al., 2025) as its backbone and applies LoRA (Hu et al., 2021) to both the visual and language components during SFT and RL. Both stages use AdamW with BF16 training. Table B.8 summarizes the principal configurations; additional optimization settings are provided in Appendices B.2 and B.3.

B.2 Supervised Fine-Tuning

SFT proceeds through Perceive, Generate, and Reason for 1,000, 1,000, and 1,500 updates, respectively. Perceive uses QA records, placing visual tokens in the input and supervising the answer block. Generate uses the full pool of visual-supervision and QA records to predict the visual-token block from the image and a feature query. Reason uses QA records to generate the complete `<think>`, `<observe>`, and `<answer>` sequence. All stages use cross-entropy with unit weight over the entire assistant target, masking user-prompt positions. Visual losses apply wherever their supervision is available; NuCLS supplies reconstruction and count supervision without cell feature alignment. Teacher targets remain fixed throughout training.

We use AdamW with a cosine learning-rate schedule, a warmup ratio of 0.05, and weight decay of 0.1. Learning rates and visual-loss weights are listed in Table B.8. A per-device micro-batch size of 1 and gradient accumulation over 24 steps across four GPUs yield an effective batch size of 96. Images are resized to 512×512, matching the resolution used for teacher feature extraction. We use random seed 0 and select the checkpoint at update 3,500 as ASPECT-SFT based on the independent development set.

Table B.8: Model and training configurations of ASPECT. SFT stage lengths are reported as updates within each stage.

Setting	Value
Model and adaptation	
Backbone	Qwen3-VL-8B
Pathology feature tokens	8
Cell tokens	6
Visual teachers	UNI; CellViT-SAM-H
Teacher updates	Frozen; targets extracted offline
LoRA-adapted components	Visual and language components
LoRA rank r	16
LoRA scaling α	32
Hardware	$4\times$ NVIDIA RTX 6000D
Supervised fine-tuning	
Total updates	3,500
Perceive / Generate / Reason updates	1,000 / 1,000 / 1,500
Effective batch size	96
LoRA learning rate	2×10^{-4}
Reconstruction and alignment module learning rate	4×10^{-5}
New-token embedding learning rate	4×10^{-5}
Count head and cell-role embedding learning rate	2×10^{-4}
Reconstruction weight λ_{u}	0.5
Cell-alignment weight λ_{a}	1.0
Count-supervision weight λ_{c}	1.0
Reinforcement learning	
Initialization	ASPECT-SFT
Algorithm	GRPO
Learning rate	1×10^{-5}
Responses per question	8
Sampling temperature	1.0
Consistency reward coefficient β	0.5
Invalid-response penalty coefficient γ	0.1

B.3 Reinforcement Learning

RL initializes from ASPECT-SFT and updates LoRA parameters in both the visual and language components using GRPO. Each rollout batch contains 48 questions, with eight responses sampled per question at temperature 1.0. Prompt and response lengths are capped at 2,048 and 512 tokens, respectively. Responses are scored using equation 6, and rewards are normalized within each question's response group to obtain relative advantages. The policy objective uses asymmetric clipping with lower and upper thresholds of 0.2 and 0.28, corresponding to a probability-ratio interval of $[0.8, 1.28]$, and a dual-clipping coefficient of 3.0. No KL penalty is included in either the reward or the optimization loss.

We use AdamW with a constant learning rate of 1×10^{-5}, no warmup, weight decay of 0.01, and random seed 42. Training runs for five epochs, with checkpoints saved every 16 updates. We select the checkpoint with the highest answer accuracy on the independent validation set described in Appendix A.4, yielding the checkpoint at update 128 used for all reported ASPECT-RL results.

B.4 Training Algorithms and Inference

We summarize the SFT and RL procedures in Algorithms 1 and 2, respectively. The algorithms specify how supervision and generated responses enter training; optimization settings are provided in Table B.8 and Appendices B.2–B.3.

Inference. ASPECT generates visual tokens, textual observations, and the final answer autoregressively from the image and question. The frozen teachers and visual-supervision modules are not

Algorithm 1 Three-stage supervised fine-tuning

Require: Backbone model; QA records $\mathcal{D}_{\mathrm{QA}}$; visual-supervision records $\mathcal{D}_{\mathrm{V}}$
Require: Cached teacher targets; stage lengths $(T_1, T_2, T_3) = (1000, 1000, 1500)$
Initialize LoRA adapters, visual-token parameters, and visual-supervision modules
for stage $s \in \{1, 2, 3\}$ **do**
  if $s = 2$ **then**
    $\mathcal{D}_s \leftarrow \mathcal{D}_{\mathrm{V}} \cup \mathcal{D}_{\mathrm{QA}}$
  else
    $\mathcal{D}_s \leftarrow \mathcal{D}_{\mathrm{QA}}$
  end if
  for $t = 1, \ldots, T_s$ **do**
    Sample a batch from $\mathcal{D}_s$
    if $s = 1$ **then** ▷ Perceive
      Input ← image, question, and visual-token block
      Target ← answer block
    else if $s = 2$ **then** ▷ Generate
      Input ← image and feature query
      Target ← visual-token block
    else ▷ Reason
      Input ← image and question
      Target ← `<think>` with visual tokens, `<observe>`, and `<answer>`
    end if
    Run a teacher-forced forward pass and collect visual-token hidden states
    Compute $\mathcal{L}_{\mathrm{LM}}$ over all target tokens, masking input positions
    Compute $\mathcal{L}_{\mathrm{uni}}$, $\mathcal{L}_{\mathrm{align}}$, and $\mathcal{L}_{\mathrm{count}}$ from available teacher targets
    Set unavailable visual-loss terms to zero
    $\mathcal{L} \leftarrow \mathcal{L}_{\mathrm{LM}} + \lambda_{\mathrm{u}}\mathcal{L}_{\mathrm{uni}} + \lambda_{\mathrm{a}}\mathcal{L}_{\mathrm{align}} + \lambda_{\mathrm{c}}\mathcal{L}_{\mathrm{count}}$
    Update trainable parameters using AdamW
  end for
end for
return ASPECT-SFT

used at inference. Reported counts are generated as part of the response, rather than read from the auxiliary count head, which provides image-wide supervision during SFT. The model determines the generated token sequence without an external routing mechanism.

C Evaluation Protocols and Metric Definitions

C.1 Benchmarks and Model Evaluation Settings

Evaluation datasets. We evaluate ASPECT and all baselines on PathoVernier and three external pathology benchmarks. For PathCLS, we use all 1,632 questions from the official PathMMU PathCLS test subset, with the original multiple-choice options and answer labels. PathMMU data are not used for SFT or checkpoint selection. For PathVQA and Quilt-VQA, we evaluate native closed-ended yes/no questions from their official test splits, retaining only H&E histology images. Each unique image is assessed independently by GPT-5.5 and Gemini-3.1-Pro and retained if either model identifies it as H&E histology. This filtering retains 1,306 of 3,362 PathVQA questions and 314 of 343 Quilt-VQA questions. All models are evaluated on the same retained subsets. These evaluations use no in-context examples; PathVQA training examples used for SFT are described in Appendix A.3.

Inference settings. Images are resized to 512×512 pixels before processing with each model's native image processor. We generate one response per question. Local models use greedy decoding with sampling disabled and seed 42; closed-source APIs use temperature zero. Output token limits are listed in Table C.9. The closed-source model identifiers are `gpt-5.5` and `gemini-3.1-pro-preview`. For PathoVernier and PathCLS, GPT-5.5 was evaluated on

Algorithm 2 Reinforcement learning with answer–observation consistency

Require: ASPECT-SFT policy π_θ; RL dataset $\mathcal{D}_{\mathrm{RL}}$; validation set $\mathcal{D}_{\mathrm{val}}$
Require: Reward coefficients β, γ; responses per question $K = 8$
Initialize the RL policy from ASPECT-SFT
for each of five training epochs **do**
  for each batch of 48 questions from $\mathcal{D}_{\mathrm{RL}}$ **do**
    for each image–question pair (x, u) with reference answer $a^\star$ **do**
      Sample responses $\{y_i\}_{i=1}^{K}$ from $\pi_\theta(\cdot \mid x, u)$
      for each response y_i **do**
        Check structural validity $V_s(y_i)$
        if $V_s(y_i) = 0$ **then**
          $R_i \leftarrow -\gamma$
        else
          Extract reported counts C_i and final answer $\hat{a}_i$
          $\mathrm{Acc}_i \leftarrow \mathbf{1}[\hat{a}_i = a^\star]$
          $\mathrm{Con}_i \leftarrow \mathbf{1}[g_u(C_i) = \pi_u(\hat{a}_i)]$, or 0 if undetermined
          $R_i \leftarrow \mathrm{Acc}_i + \beta\,\mathrm{Con}_i$
        end if
      end for
      Normalize group rewards to obtain relative advantages
    end for
    Update visual and language LoRA parameters with GRPO
    Use asymmetric clipping and no KL penalty as specified in Appendix B.3
    Save a checkpoint every 16 training steps
  end for
end for
return The saved checkpoint with the highest accuracy on $\mathcal{D}_{\mathrm{val}}$

Table C.9: Evaluation subsets and output token limits for local models and closed-source APIs.

Benchmark	Evaluation subset	Questions	Local token limit	API token limit
PathoVernier	Full benchmark	759	512	512
PathCLS	Official test	1,632	512	512
PathVQA	H&E, closed-ended test	1,306	512	512
Quilt-VQA	H&E, closed-ended test	314	512	512

September 2, 2026, and Gemini-3.1-Pro on September 3, 2026. Both models were evaluated on PathVQA and Quilt-VQA on September 15, 2026.

Prompts and scoring. On PathoVernier, all models receive identical questions and answer options. Baselines additionally receive a system prompt requesting whole-image cell-type counts, the quantities required by the question, and a final answer, whereas ASPECT uses its trained response format without a system prompt. All responses are scored with the same parser. PathVQA and Quilt-VQA request only yes/no answers and use no system prompt. PathCLS requests a brief morphological justification followed by the selected option letter. Intermediate counts are not required for these external tasks. Unparseable final answers are scored as incorrect. Failed API requests are retried up to four times, and unresolved failures are also scored as incorrect. Metric definitions and eligibility criteria are detailed in Appendix C.2, and the evaluation prompts are provided in Appendix D.4.

C.2 Task Rules, Metrics, and Eligible Responses

Task rules. Reference counts are obtained by assigning annotated nuclei to the geometric regions specified in each question according to their centroids. Region selection identifies the region with the largest count of the queried cell type among equal-area regions. Region comparison compares

one cell type across two equal-area regions, while cell-type comparison compares two types within the image. Let a and b denote the first and second counts in the comparison. For $b > 0$, both tasks assign the ratio $r = a/b$ to a category using

$$h(r) = \begin{cases} \texttt{much_fewer}, & r \leq 0.5, \\ \texttt{comparable}, & 0.5 < r < 2.0, \\ \texttt{much_more}, & r \geq 2.0. \end{cases} \tag{11}$$

When $b = 0$, the result is `much_more` if $a > 0$ and `comparable` if $a = 0$. These cases remain determinable. For equal-area regions, the count ratio equals the density ratio.

Multi-step composition first selects the region with the largest count of the queried type and then assigns that type's proportion p within the selected region to a category:

$$h_{\text{prop}}(p; t_1, t_2) = \begin{cases} \texttt{low}, & p < t_1, \\ \texttt{mid}, & t_1 \leq p < t_2, \\ \texttt{high}, & p \geq t_2. \end{cases} \tag{12}$$

The thresholds t_1 and t_2 are estimated from training-split tertiles separately for each cell type and spatial partition, rounded to increments of 0.05, and then frozen. Each question states its applicable thresholds explicitly. Table C.10 lists the thresholds for quadrants, four horizontal bands, and four vertical bands. Tumor is excluded from this task because its median proportion is approximately one. Candidate questions whose reference proportions fall within a 5% relative margin of either threshold are excluded during construction.

Table C.10: Frozen proportion thresholds (t_1, t_2) for multi-step composition, estimated separately for each spatial partition and cell type.

Partition	Stromal-like	Lymphocyte	Epithelial-like	Inflammatory
Quadrants	$(0.50, 0.85)$	$(0.35, 0.55)$	$(0.60, 0.90)$	$(0.55, 0.80)$
Horizontal bands	$(0.40, 0.80)$	$(0.30, 0.50)$	$(0.55, 0.80)$	$(0.55, 0.75)$
Vertical bands	$(0.40, 0.75)$	$(0.30, 0.50)$	$(0.60, 0.85)$	$(0.55, 0.75)$

If reported counts yield a tied maximum in region selection or the first step of multi-step composition, the count-derived region is undeterminable. Such responses are excluded from CA and Consistency, but remain eligible for Acc and Count Acc; RAWR eligibility follows its correct-answer and complete-count requirements. A tied maximum therefore differs from a zero denominator in a comparison task, for which the rule above returns a definite category.

Answer and count accuracy. Let $\hat{a}$ and $a^\star$ denote the predicted and reference answers, respectively. Answer accuracy evaluates the complete answer, including both the region and proportion category for multi-step composition. An unparseable final answer is scored as incorrect. To evaluate the reported measurements, let K index the counts required by a question, q_i denote reference count i, and $\text{dom}(C)$ denote the reported count keys. The per-question count score is

$$S(C, Q) = \frac{1}{|K|} \sum_{i \in K \cap \text{dom}(C)} \mathbf{1}[|C_i - q_i| \leq \max(1, 0.1 q_i)] \,, \tag{13}$$

where $Q = \{q_i\}_{i \in K}$ is the set of required reference counts and $\mathbf{1}[\cdot]$ is the indicator function. Each required count contributes equally, and missing counts contribute zero. We report

$$\text{Acc} = \mathbb{E}[\mathbf{1}[\hat{a} = a^\star]] \,, \qquad \text{Count Acc} = \mathbb{E}[S(C, Q)] \,. \tag{14}$$

Both averages include all evaluation questions, with equal weight per question. Count Acc therefore measures the fraction of required counts within tolerance rather than exact integer-count accuracy.

Gold-referenced coherent accuracy. CA evaluates whether the reported counts imply the reference task conclusion. Let $g_u(C)$ denote the conclusion checked by CA and $\pi_u(a^\star)$ the corresponding component of the reference answer:

$$\text{CA} = \mathbb{E}[\mathbf{1}[g_u(C) = \pi_u(a^\star)] \,|\, g_u(C) \text{ is defined}] \,. \tag{15}$$

For region selection and the two comparison tasks, this check uses the complete task conclusion. For multi-step composition, g_u checks only region selection, and π_u extracts the reference region. Its eligibility therefore depends on the counts needed to select a region, without requiring the additional counts needed to determine the proportion category. CA does not require the model's final answer to be correct. Moreover, counts can imply the correct conclusion while remaining numerically inaccurate, so CA and Count Acc measure different properties.

Right answer, wrong reason. RAWR evaluates counting errors among responses that have a correct final answer and report every required count:

$$\mathrm{RAWR} = 1 - \mathbb{E}[S(C, Q) \mid \hat{a} = a^\star,\ K \subseteq \mathrm{dom}(C)]\,. \tag{16}$$

Thus, Count Acc averages count agreement over all questions, whereas RAWR averages the fraction of counts outside tolerance over correct responses with complete measurements. RAWR is not a binary indicator of whether a response contains any counting error.

Consistency with the model's own answer. Consistency evaluates whether the conclusion implied by the reported counts agrees with the corresponding component of the model's own answer. Using the same task check g_u and answer projection π_u as CA, we define

$$\mathrm{Consistency} = \mathbb{E}[\mathbf{1}[g_u(C) = \pi_u(\hat{a})] \mid g_u(C) \text{ and } \pi_u(\hat{a}) \text{ are defined}]\,. \tag{17}$$

CA compares the count-derived conclusion with $\pi_u(a^\star)$, whereas Consistency compares it with $\pi_u(\hat{a})$. For multi-step composition, both checks evaluate region selection only and do not require the proportion denominator. The full region-and-proportion answer is evaluated by Acc, while $S(C, Q)$ measures agreement between reported and reference counts. Consequently, a response can be consistent despite an incorrect proportion category or an incorrect final answer. The indicator in equation 17 is the same as Con in the RL reward: undeterminable cases receive zero consistency reward during training but are excluded from the evaluation metric's denominator. The off-diagonal categories in Figure 5 therefore do not measure the rate of inconsistency with the model's own answer.

Eligible responses. Acc and Count Acc include all evaluation questions. CA includes responses with a determinable count-derived conclusion even when the final answer is missing or unparseable. RAWR requires a correct final answer and complete reported counts, whereas Consistency requires both the count-derived conclusion and the corresponding predicted-answer component to be determinable. Counts are scored field by field using the evaluation parser's output; negative values or duplicate keys do not automatically assign zero to the entire response. The training validity gate V_s is not applied as a response-level filter during evaluation. Missing required counts contribute zero to Count Acc. Count coverage is the fraction of all 759 questions for which the reported counts yield a determinable conclusion under the task rules used for CA. For multi-step composition, this requires determining the selected region, irrespective of the proportion band. Complete-count coverage instead requires every necessary count to be reported, regardless of answer correctness. Complete counts can still yield an undeterminable conclusion when regional maxima are tied. In Table 1, CA and RAWR are omitted when CA coverage is below 30%; this reporting threshold does not change either metric's definition or eligible sample set. Table C.11 reports both coverage measures, RAWR sample counts, and unconditional Count Acc.

ASPECT achieves the highest Count Acc over all questions (0.471 versus 0.296 for Gemini-3.1-Pro), showing that its improvement in measurement accuracy extends beyond the conditional subsets used for CA and RAWR.

C.3 Ablation and Image-Mismatch Protocols

Visual supervision and training curriculum. We evaluate the SFT ablations before RL to separate the effects of supervised training from reward optimization. Starting from the same backbone, we remove pathology feature reconstruction, cell feature alignment, or count supervision individually, and additionally evaluate a variant without all three visual losses. These loss-removal variants retain the visual tokens, response formats, three-stage curriculum, and remaining loss weights. Assistant-token cross-entropy has weight one throughout training. Direct Reason retains all three visual losses and trains on the 13,095 QA examples using the Reason-stage response format from the first update. It runs for 3,500 updates, matching the total update budget of the complete curriculum.

Table C.11: Evaluation coverage and unconditional count accuracy on PathoVernier. Both coverage measures report counts out of all 759 questions. N_{RAWR} is the number of correct responses with complete reported counts. Count Acc averages $S(C, Q)$ over all questions, including failures and incomplete responses.

Model	CA coverage	Complete-count coverage	N_{RAWR}	Count Acc↑
GPT-5.5	748/759	749/759	447	0.210
Gemini-3.1-Pro	733/759	750/759	472	0.296
Qwen3-VL-8B	233/759	261/759	106	0.068
Qwen3-VL-32B	682/759	700/759	333	0.178
InternVL3-8B	448/759	449/759	166	0.071
InternVL3-38B	750/759	753/759	340	0.153
Quilt-LLaVA-7B	1/759	2/759	1	0.000
PathGen-LLaVA-13B	0/759	0/759	0	0.000
HuatuoGPT-Vision-7B	12/759	12/759	2	0.002
Patho-R1-7B	0/759	0/759	0	0.000
ASPECT-8B (SFT)	714/759	738/759	525	0.444
ASPECT-8B (RL)	748/759	752/759	563	**0.471**

Unlike the full curriculum, whose Generate stage also includes visual-supervision examples, Direct Reason uses QA examples throughout. All variants follow the evaluation protocol in Appendix C.1.

Reinforcement learning rewards. The reward ablations initialize from the same ASPECT-SFT checkpoint and use the same 1,372-question RL set and GRPO configuration. We retain the structural validity gate and invalid-response penalty in every variant:

$$R = V_s\, r - 0.1(1 - V_s), \tag{18}$$

where the valid-response reward r is Acc, $0.5\,\mathrm{Con}$, $\mathrm{Acc}+0.5\,S(C, Q)$, or $\mathrm{Acc}+0.5\,\mathrm{Con}$ for answer-only, consistency-only, answer-plus-count, and ASPECT RL, respectively. The count-reward variant uses the per-question tolerance score defined in equation 13, rather than the dataset-level Count Acc. The SFT row provides the initialization baseline without RL. Checkpoints are selected by answer accuracy on the independent validation set.

Image mismatch. Figure 5(a) evaluates the model without auxiliary visual supervision, ASPECT-SFT, and ASPECT-RL using original and mismatched images. We construct a permutation of unique patches within each source dataset, requiring every patch to map to a different patch. This preserves dataset provenance while breaking the correspondence between the image and question. All questions associated with an original patch share the same replacement image, and all three models use identical mappings. We generate three mappings with seeds 0, 1, and 2 over the initial 786-question candidate set and evaluate only the final 759 benchmark questions. Questions, options, reference answers, and reference counts remain unchanged. We report the mean mismatched-image Acc and Count Acc across the three mappings; all corresponding standard deviations are at most 0.017. This variation reflects image replacements rather than training runs.

Joint outcome analysis. Figure 5(b) decomposes the responses of ASPECT-SFT, answer-only RL, and ASPECT-RL according to final-answer correctness and the gold-referenced count check. For each response, we define

$$A = \mathbf{1}[\hat{a} = a^\star], \qquad B = \mathbf{1}[g_u(C) = \pi_u(a^\star)], \tag{19}$$

where g_u and π_u follow Appendix C.2. Responses with a determinable count-derived conclusion are assigned to the four (B, A) combinations; responses for which $g_u(C)$ is undeterminable form a fifth category. All category proportions use the full 759-question benchmark as their denominator. For multi-step composition, B evaluates region selection, whereas A evaluates the complete region-and-proportion answer. Thus, $B = 1, A = 0$ includes responses that select the correct region but give an incorrect proportion category. This group need not be inconsistent with the model's own reported counts, and the off-diagonal proportion is not the complement of Consistency.

D PROMPTS AND RESPONSE FORMATS

D.1 TRAINING DATA GENERATION

We generate the textual content of SFT question-answer records using GPT-5.6-sol. Each request includes the H&E image, question, reference answer, and source-specific context. The five nucleus-annotated datasets provide reference cell counts, whereas cellular NCT-CRC-HE-100K patches provide pseudo-counts together with their tissue labels. PathVQA supplies the original yes/no answer without a nuclear composition input. All sources share the system prompt below. The displayed prompts are excerpts, and braces denote sample-specific fields.

GPT-5.6-sol: Shared system prompt

You are an expert pathologist. You are given an H&E image, the GROUND-TRUTH nuclear composition of the field, a question, and the CORRECT answer. Your job is NOT to decide the answer (it is already given and correct).
Produce EXACTLY two lines:
`OBSERVE:` one sentence describing the overall visual appearance of the field — cellularity, tissue architecture, dominant nuclear morphology — WITHOUT stating any numeric counts (no digits).
`REASON:` 2-3 sentences that justify the GIVEN answer, citing counts ONLY from the provided composition, leading to that answer.
Rules: never invent numbers; OBSERVE must contain NO digits; do NOT repeat the OBSERVE sentence inside REASON . . . ; do NOT write 'FINAL:' or the answer token — that is appended automatically.

The source-specific user inputs are summarized below. For the five nucleus-annotated datasets, the composition includes explicitly supplied zero counts for absent types. CRC inputs distinguish cellular patches, which supply detected composition, from non-cellular patches. PathVQA instead requests an explanation based on image morphology.

GPT-5.6-sol: Source-specific user inputs

Five nucleus-annotated datasets
Tissue/organ: {`organ`} (H&E).
Ground-truth nuclear composition of THIS field: {`composition`}.
Question: {`question`}
CORRECT answer: {`gold`}
Output the two lines now:
`OBSERVE: ...`
`REASON: ...`

NCT-CRC-HE-100K
H&E colorectal histology.
Ground-truth tissue class: {`option`} ({`tissue_class`}).
For cellular patches:
Detected nuclear composition: {`composition`}.
For non-cellular patches:
This field is essentially acellular (no significant nuclei).
Question: {`question`}
CORRECT answer: {`gold`}
Output the two lines now:
`OBSERVE: ...`
`REASON: ...`

PathVQA
H&E pathology microscopy image.
Question: {`question`}
CORRECT answer: {`yes/no`}
Write two lines. OBSERVE = one sentence on the overall appearance relevant to the question (no digits). REASON = 1-3 sentences that justify the given yes/no answer from the morphology.
`OBSERVE: ...`
`REASON: ...`

The generated lines are incorporated into the <observe> and <answer> blocks. Where count observations are included, their JSON is inserted programmatically from the corresponding annotations or pseudo-labels. PathVQA observations contain qualitative descriptions without count JSON. The FINAL: line is appended from the reference answer.

For counting questions, GPT-5.6-sol subsequently rewrites the reasoning notes in <answer>, including their count statements, conclusions, and final-answer lines. It receives text only, with multiple notes grouped by identifiers. Rewrites must preserve the numerical multiset and the final-answer line; failures revert to the template version.

GPT-5.6-sol: Answer-note rewriting

You are rewriting the wording of short analytical notes that accompany histopathology counting questions. Each note states some counts and draws a conclusion. Rewrite EACH note below.
Hard requirements:

- Keep every number EXACTLY as given. Do not add, drop, round or recompute any number. You cannot see the image, so you have no basis to change any count.
- Keep the final line exactly as FINAL: <option> with the option unchanged.
- Vary sentence structure, connectives and register . . . ; do not reuse one skeleton with different numbers . . .
- Keep it to one or two sentences plus the FINAL line.
- Output . . . one rewritten note per input, each introduced by ### <id> . . .

Notes to rewrite:
{id}
{answer_note_including_FINAL}
. . .

D.2 Benchmark Question Rewriting

We use GPT-5.6-sol to diversify question wording while preserving the queried quantities. Training and benchmark questions use separate phrasing pools, denoted A and B, with no template shared between them. The generator receives a reference template, its placeholder list, and an explicit description of the intended measurement. Numerical rules, reference counts, and final answers are handled programmatically rather than generated by the paraphrasing model.

GPT-5.6-sol: Question paraphrasing

You are helping diversify the phrasing of questions in a computational-pathology benchmark.
Reference question template (Python format string): {ref}
Placeholders: {ph}
What the question asks: {meaning}
Write {n} ALTERNATIVE phrasings of the SAME question. Hard requirements:

- Use EXACTLY the same placeholders, each exactly once. Do not add or drop any.
- Ask for exactly the same quantity. Do not change what is being compared, which region, or how it is normalised.
- Do NOT hint at the answer: the question must not contain comparative or magnitude words such as more/fewer/higher/lower/most/least/high/low/dense.
- Vary sentence structure and register (some clinical, some plain), not just synonyms.
- One per line, no numbering, no quotes, nothing else.

Here, {ref} is the reference template, {ph} lists its placeholders, {meaning} specifies the quantity being queried, and {n} is the requested number of alternatives. Within question templates, {t} denotes the queried nucleus type, {t1} and {t2} denote two compared types, and {r1} and {r2} denote two compared regions. The placeholder {region} denotes the whole-image field. The following example specifies that both steps of a multi-step composition question concern the same nucleus type.

Example generator input: Multi-step composition

Reference template
Determine the region whose {t} nuclear density is greatest, then state the proportion of that region's nuclei that are {t}.
Placeholder list
{t}
Intended measurement
first which equal-area region contains the greatest density of nucleus type {t}, then what proportion of ALL nuclei in THAT region are of the SAME type {t}. Only one nucleus type is involved throughout — do not introduce a second type.

Candidates first undergo deterministic checks for placeholder identity and prohibited directional wording. Geometric-region questions must retain the word "region", use "nuclei" rather than "cells", and restrict "area" to the expression "per unit area". Cross-pool Jaccard similarity must not exceed 0.55. Candidates passing these checks are evaluated by GPT-5.4, using the reference and candidate templates instantiated with the same placeholder values.

GPT-5.4: Independent semantic verification

Two questions about the same histopathology image are shown.
A: {a}
B: {b}
Do A and B ask for EXACTLY the same quantity — same object type, same region, same normalisation (per-area vs raw count), same comparison direction? A difference in wording is fine; a difference in what is measured is not.
Answer with ONLY one word: SAME or DIFFERENT.

The candidate is accepted when the verifier returns SAME and rejected when it returns DIFFERENT. Failed candidates are regenerated, with the original template retained when retries are exhausted. Rewriting changes question wording without modifying the numerical keys in <observe> or the programmatically determined FINAL answer. Independent answer verification and expert review follow the procedures in Appendix A.5.

D.3 SFT AND RL PROMPTS

We present the user inputs and assistant targets used in the three SFT stages, followed by the prompt used for RL. Images precede text in every input. For compactness, [UNI block] denotes <|anchor_start|>, eight consecutive <|uni_pad|> tokens, and <|anchor_end|>; [CELL block] uses the same delimiters with six consecutive <|cell_pad|> tokens. These bracketed names are display abbreviations and are replaced by the corresponding token sequences in training. The UNI block always precedes the CELL block. SFT applies cross-entropy to all assistant-target tokens with unit weight and masks user-prompt positions.

Perceive: Answer from visual tokens

User input
<image>the patch feature of the image is [UNI block], and the cell composition of the image is [CELL block]. {QUESTION}

Assistant target
{ANSWER}

Here, {ANSWER} is the prepared answer block, including its <answer> and </answer> delimiters. The target contains neither a <think> block nor an <observe> block. Visual tokens occur in the user input and therefore receive visual supervision but no token-prediction loss at their input positions.

Generate: Predict visual tokens

User input
<image>What is the patch feature, cell composition of the image?

Assistant target
[UNI block][CELL block]

Generate uses both visual-supervision records and QA records, replacing their questions with the feature query above. Every record supervises prediction of the visual-token sequence through cross-entropy, together with the available visual supervision losses.

Reason: Generate observations and an answer

User input
<image>{QUESTION}

Assistant target
<think> the patch feature of the image is [UNI block], and the cell composition of the image is [CELL block]. </think>
<observe>{OBSERVATION}</observe>
<answer> {REASONING}
FINAL: {OPTION} </answer>

The Reason target places the visual tokens inside the <think> block before the observation and answer. The {OBSERVATION} field contains the descriptive sentence and, where provided by the data source, the count JSON described in Appendix D.1. PathVQA observations contain only descriptive prose. The question includes its answer options.

Reinforcement learning: Sample a response

System prompt
None.

User input
<image>
{QUESTION}
Choose exactly one of: {opt1, opt2, ...}

Assistant response
Sampled from the policy; no fixed target is supplied.

RL retains the image-and-question input structure of Reason and explicitly lists the available options. The policy generates its response in the format learned during SFT, without additional formatting instructions in the prompt. Responses are scored using the reward described in Appendix B.3.

For training samples marked as non-cellular, the CELL block and the corresponding cell-composition clause are omitted. Their Generate query is "What is the patch feature of the image?", and its target contains only the UNI block. This adjustment applies to training examples; inference uses no external routing to select which visual-token blocks the model generates.

D.4 Evaluation Prompts

We provide the evaluation prompts below. Each question is accompanied by its image; image inputs are omitted from the displayed templates. Placeholders are replaced with the question and its available answer options. All evaluations use zero-shot prompting without demonstration examples.

PathoVernier. Baseline models receive explicit instructions to report nuclear counts before their final answer. ASPECT uses no system prompt and follows the response format learned during SFT, with the image, question, and available options as input, as illustrated in Appendix D.3. All models receive the same questions and answer options.

PathoVernier: Baseline evaluation

System prompt
You are an expert pathologist analyzing an H&E-stained histology image. You MUST base your answer strictly on what is visible in THIS image: actually look at and count the nuclei you can see. Never answer from priors, definitions, or the wording of the question alone — if you did not look at the image, you cannot answer. Think step by step. Then output ONE JSON object wrapped in `<observe>...</observe>`: FIRST the whole-image count of each main nucleus type you can see — epithelial-like, stromal-like, lymphocyte, inflammatory, and tumor separately when identifiable (do NOT report a type that is not present in this patch) — THEN the specific quantities the question asks you to report. Finally, output a line 'FINAL: X' where X is EXACTLY one of the allowed option tokens, copied verbatim.

User prompt
`{question}`, Choose exactly one of these option tokens: `{opt1, opt2, ...}`. Respond with your reasoning, then `<observe>{...}</observe>`, then a final line '`FINAL: <token>`'.

PathCLS. The prompt asks the model to identify the tissue or lesion, briefly justify its choice from the image, and end with the selected option letter. It does not require structured count observations.

PathCLS: Tissue and lesion classification

System prompt
You are an expert pathologist analyzing a single H&E-stained histology image. This is a MULTIPLE-CHOICE QUESTION about which tissue or lesion type the image shows. You may first note what you observe about the cells, but your final `<answer>` MUST directly address THIS question: decide which of the listed options best matches the tissue/lesion in the image, briefly justify it from the morphology, and end with a line 'FINAL: X' where X is EXACTLY one option letter. Do NOT just describe cell counts — you must pick the correct option.

User prompt
`{question}`
Options: A) ... B) ...
In your answer, choose the single option that correctly names the tissue/lesion shown, and end with '`FINAL: <letter>`'.

PathVQA and Quilt-VQA. Both benchmarks use the same user prompt without a system prompt. Models are asked to return only the binary answer, without reasoning or count observations.

PathVQA and Quilt-VQA: Closed-ended questions

System prompt
None.

User prompt
You are looking at a histopathology image.
`{question}`
Answer with exactly one word: yes or no.

E Additional Quantitative Results

E.1 Results by Task Type

Figure E.1 compares ASPECT with its base model and two closed-source models across the four PathoVernier task types. ASPECT achieves higher Acc and lower RAWR in every task, extending its advantage from region selection to multi-step composition and quantitative comparisons. Relative to Gemini-3.1-Pro, its Acc improves by 31.3% on multi-step composition and 26.3% on cell-type comparison. Lower RAWR across all four tasks further shows that, among correct answers with complete count observations, ASPECT reports more accurate quantitative evidence. These results connect its gains in task accuracy with improved measurement of the underlying cell composition.

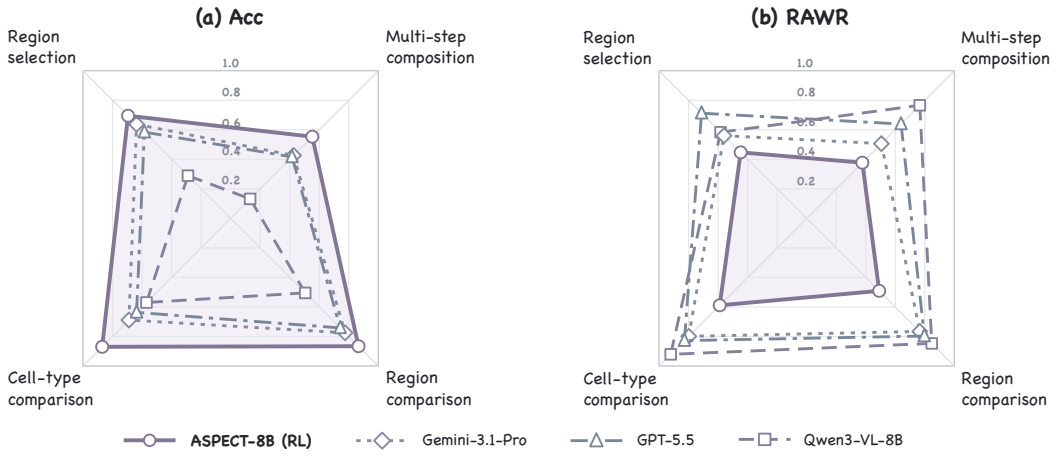


Figure E.1: Results by task type on PathoVernier. ASPECT achieves higher Acc and lower RAWR than the three comparison models across all four tasks. Larger values are better for Acc; smaller values are better for RAWR.

(a) Acc

	Lizard 0.50	PUMA 0.22	PanNuke 0.25	CoNSeP 0.25	NuCLS 0.20
ASPECT-8B (RL)	**0.820**	**0.669**	**0.647**	**0.659**	**0.706**
Qwen3-VL-8B	0.400	0.345	0.410	0.250	0.206
Gemini-3.1-Pro	0.705	0.592	0.511	0.477	0.500
GPT-5.5	0.583	0.662	0.554	0.568	0.647

0 0.25 0.5 0.75 1

(b) RAWR

	Lizard 0.50	PUMA 0.22	PanNuke 0.25	CoNSeP 0.25	NuCLS 0.20
ASPECT-8B (RL)	**0.538** n=326	**0.364** n=94	**0.400** n=90	**0.474** n=29	0.656 n=24
Qwen3-VL-8B	0.855 n=55	0.886 n=22	0.821 n=21	-- n=4	-- n=4
Gemini-3.1-Pro	0.738 n=280	0.584 n=83	0.574 n=71	0.679 n=21	**0.632** n=17
GPT-5.5	0.816 n=231	0.673 n=94	0.701 n=77	0.696 n=23	0.739 n=22

1 0.75 0.5 0.25 0

Pixel sizes below source names are in micrometers per pixel; gray RAWR cells have fewer than 10 eligible questions.

Figure E.2: Results by source dataset on PathoVernier. Column labels include native pixel sizes in μm/pixel. Darker shading indicates better performance within each panel. In the RAWR panel, n denotes the number of correct answers with complete count observations; estimates with $n < 10$ are withheld. Source and imaging scale are evaluated jointly.

E.2 Results by Data Source and Imaging Scale

Figure E.2 breaks down performance across the five source datasets and their native pixel sizes. ASPECT achieves the highest Acc on every source, showing that its gains extend beyond Lizard, which contributes more than half of PathoVernier. On PUMA, ASPECT and GPT-5.5 obtain similar Acc (0.669 versus 0.662), yet ASPECT achieves substantially lower RAWR (0.364 versus 0.673). This comparison illustrates how similar answer accuracy can conceal differences in the quality of reported counts. ASPECT achieves the lowest RAWR on four sources; NuCLS is the exception, where Gemini-3.1-Pro has lower RAWR (0.632 versus 0.656) despite lower Acc. The source-level results thus reveal both the broad improvement in task accuracy and the variation in the accuracy of its supporting measurements.

E.3 Statistical Uncertainty

We quantify test-set sampling uncertainty using 10,000 paired bootstrap resamples of the 553 PathoVernier patches, stratified by source dataset, with random seed 42. Within each source, we sample its original number of patches with replacement and retain all associated questions, includ-

ing sampling multiplicities. All models share the same resamples. We recompute each metric and its denominator according to Appendix C.2, retaining API failures as incorrect answers for Acc and using each model's own eligible samples for CA and RAWR. We report the 2.5th and 97.5th percentiles as 95% confidence intervals.

As shown in Table E.12, all six paired improvement intervals lie above zero. Against Gemini-3.1-Pro, ASPECT improves Acc by 0.120, with a 95% CI of [0.077, 0.162]. The positive intervals for CA gains and RAWR reductions further support improvements in both count-derived conclusions and the reported quantities behind correct answers. These comparisons use the paired difference distributions, rather than overlap between individual model intervals.

Table E.12: Model scores and paired improvements on PathoVernier with 95% bootstrap confidence intervals. Improvements are ASPECT minus baseline for Acc and CA, and baseline minus ASPECT for RAWR; positive values favor ASPECT.

Model	Acc↑	CA↑	RAWR↓
Qwen3-VL-8B	0.374 [0.341, 0.408]	0.532 [0.471, 0.595]	0.854 [0.801, 0.901]
Gemini-3.1-Pro	0.626 [0.588, 0.663]	0.715 [0.681, 0.749]	0.680 [0.651, 0.708]
GPT-5.5	0.594 [0.558, 0.631]	0.652 [0.616, 0.689]	0.756 [0.728, 0.784]
ASPECT-8B	**0.746** [0.714, 0.776]	**0.807** [0.777, 0.836]	**0.489** [0.460, 0.518]
Comparison	Acc gain	CA gain	RAWR reduction
ASPECT vs Qwen3-VL-8B	+0.372 [+0.327, +0.415]	+0.275 [+0.207, +0.342]	+0.365 [+0.305, +0.420]
ASPECT vs Gemini-3.1-Pro	+0.120 [+0.077, +0.162]	+0.092 [+0.052, +0.132]	+0.191 [+0.152, +0.229]

E.4 Sensitivity to Count Tolerance

We vary the relative count tolerance r over $\{0, 0.05, \ldots, 0.30\}$ while retaining the absolute tolerance floor of one nucleus. A reported count c agrees with its reference count q when

$$|c - q| \leq \max\{1, rq\}. \tag{20}$$

We recompute the count-agreement score and RAWR at each tolerance, keeping each model's correct-answer subset with complete count observations fixed. The main results correspond to $r = 0.10$, rather than an average over this tolerance grid. At $r = 0$, the criterion still permits an absolute counting error of one.

Figure E.3 shows that ASPECT achieves the lowest RAWR at all seven tested tolerances. Even under the strictest setting, its RAWR is 0.578, compared with 0.725 for Gemini-3.1-Pro. Their gap increases from 0.147 at $r = 0$ to 0.256 at $r = 0.30$. The consistent ranking across the scan shows that ASPECT's advantage in reported count accuracy extends beyond the default tolerance.

F Complete Case Studies and Failure Analysis

F.1 Complete Responses for Main-Text Examples

Figures G.4 and G.5 provide the complete responses for the two examples discussed in the main-text qualitative comparison. Each figure includes the question, reference answer, and outputs from ASPECT, Qwen3-VL-8B, Gemini-3.1-Pro, and Patho-R1-7B.

PathoVernier: Correct answers and their quantitative basis. The reference answer selects the lower-right region and assigns a low lymphocyte proportion, based on 7 lymphocytes among 29 nuclei. Gemini reaches the correct answer using inaccurate counts of 12 and 47: their ratio falls in the same band despite the counting errors. Qwen3-VL-8B also selects the correct region, but its estimate of 20 out of 50 nuclei leads to the incorrect mid band. ASPECT recovers the reference counts for the selected region and derives the correct low band from $7/29 \approx 0.24$. This example illustrates why final-answer accuracy alone cannot distinguish accurate measurements from counting errors that preserve the answer category.

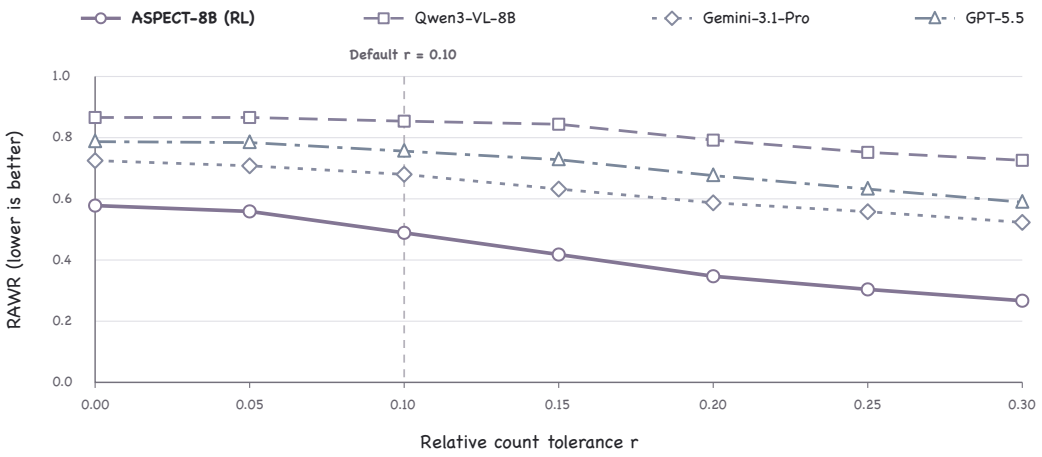


Figure E.3: RAWR under varying relative count tolerance r, with an absolute tolerance floor of one nucleus. The dashed line marks the main evaluation setting, $r = 0.10$. Each model's eligible sample set remains fixed throughout the scan. Lower is better.

PathCLS: Tissue and lesion classification. For the skin histology example, ASPECT selects the reference class, squamous cell carcinoma (N), while Qwen3-VL-8B selects basal cell carcinoma (M) and Gemini selects melanoma (O). Patho-R1 describes non-tumor epidermis in its reasoning but outputs K, which denotes non-tumor sweat glands. The complete responses expose differences in both class selection and the correspondence between the written rationale and the selected option.

F.2 Additional PathoVernier Examples

Figures G.6–G.8 supplement the multi-step example in Appendix F.1 with region comparison, cell-type comparison, and region selection. The complete responses show how count estimates and their subsequent interpretation affect the final answer.

Region comparison. In Figure G.6, the reference tumor counts are 15 in the top half and 2 in the bottom half, giving the answer `much_more`. ASPECT reports 16 and 3 and reaches the same comparison category. Gemini reports 12 and 8, leading to the incorrect `comparable` answer. Qwen3-VL-8B exhibits a further discrepancy: its reasoning uses $6.5/3.5$, but its final reported counts are 6 and 3. The latter imply `much_more` at the stated threshold, although its final answer remains `comparable`.

Cell-type comparison. Figure G.7 asks whether stromal and inflammatory nuclei have comparable abundance. The reference counts are 57 and 44. Gemini estimates 2 and 65 and answers `much_fewer`, whereas Qwen3-VL-8B reports 17 and 6 and answers `much_more`. ASPECT reports 54 and 50 and correctly selects `comparable`. The opposing baseline conclusions arise from markedly different estimates of the same two cell populations.

Region selection. In Figure G.8, the reference stromal counts across four equal-area horizontal bands are $(16, 4, 1, 1)$, making the topmost band, `r1`, the correct answer. ASPECT reports $(16, 5, 2, 1)$ and selects `r1`, recovering the dominant concentration in the top band. Patho-R1's reasoning also points to `r1`, although its reported counts of $(2, 0, 0, 1)$ substantially underestimate the stromal population. Gemini and Qwen3-VL-8B instead select `r2` and `r4`, respectively. This example further separates correct regional selection from accurate measurement.

F.3 Failure Analysis

Figure G.9 presents two ASPECT failures in which the final answers follow the reported counts, but those counts misrepresent the regional composition.

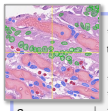
Incorrect regional distribution despite a near-correct total. In the CoNSeP example (right), ASPECT reports 47 tumor nuclei across the image, close to the reference total of 46. Its regional counts, however, are $(0, 4, 7, 36)$ rather than $(0, 8, 28, 10)$. Selecting the largest reported count consequently yields `r4` instead of the reference region `r3`. The response is internally consistent, but the near-correct total does not ensure accurate regional measurement. Across 43 failed strip-based density-selection questions, 27 (62.8%) select a band adjacent to the reference band.

Denominator overestimation after correct region selection. In the Lizard example (left), ASPECT correctly identifies the upper-right quadrant and estimates 19 stromal nuclei, close to the reference count of 18. It nevertheless reports 53 total nuclei in that region instead of 31, reducing the estimated stromal proportion from the reference $18/31 \approx 0.58$ to $19/53 \approx 0.36$. This changes the answer from `mid` to `low` at the 0.50 threshold. The case passes the region-only CA check but fails Acc, which evaluates the complete region-and-band answer. Among 44 multi-step questions with correct region selection but an incorrect proportion band, the median predicted-to-reference denominator ratio is 1.72, close to this example's $53/31 \approx 1.71$.

G Limitations and Future Work

Spatial and clinical scope. This study focuses on quantitative visual reasoning within H&E image patches. Pathological interpretation also draws on tissue architecture over larger fields, relationships between spatially separated regions, and patient-level context. Extending the present framework to whole-slide analysis would require coordinating local measurements with region selection and broader tissue organization. An important direction is to study how verifiable observations at different spatial scales can support slide-level assessments and clinically relevant endpoints.

Task and annotation scope. PathoVernier evaluates cell composition through a shared taxonomy and questions with explicit quantitative reference answers. This formulation makes intermediate measurements assessable across datasets, while its scope remains tied to the categories and quantities supported by the source annotations. Morphological attributes, interactions between cell populations, and richer descriptions of tissue organization offer complementary dimensions of pathology understanding. Future work could develop reference annotations and evaluation criteria for these dimensions, extending the connection between visual evidence and model conclusions beyond the count-based tasks studied here.



Source: PathoVernier

Question: Determine the region whose lymphocytic nuclear density is greatest, then state the proportion of that region's nuclei that are lymphocytic. For the share, use these bands: 'low' below 0.35, 'high' at or above 0.55, otherwise 'mid'.

GT per-region lymphocyte counts (asked = lymphocyte): `{'upperleft': 2, 'upperright': 0, 'lowerleft': 2, 'lowerright': 7}`
GT lower-right region reference: lymphocyte `_num=7` / **total nuclei `_den=29`** → lymphocyte share = 7/29 ≈ 0.24 → band `low` (< 0.35)
Gold answer: `lowerright:low` · options: ['lowerright:low', 'lowerleft:high', 'upperleft:mid', 'lowerright:high', 'lowerright:mid', 'upperright:high', 'lowerleft:low', 'lowerleft:mid', 'upperright:mid', 'upperleft:high', 'upperright:low', 'upperleft:low']

Gemini-3.1-Pro

To determine the region with the highest lymphocytic nuclear density, we first divide the image into four quadrants and count the lymphocytes (small, dark, round nuclei) in each.
1. **Upper-left (UL):** This quadrant contains mostly empty space and some large tumor cells. There is approximately 1 lymphocyte visible.
2. **Upper-right (UR):** This quadrant is densely packed with large tumor cells. There are about 2 lymphocytes visible.
3. **Lower-left (LL):** This quadrant contains tumor cells, pink keratin/osteoid material, and a scattering of lymphocytes. We can count about 8 lymphocytes here.
4. **Lower-right (LR):** This quadrant contains many tumor cells and a noticeable cluster of lymphocytes, particularly in the center-left and bottom areas of the quadrant. We can count about 12 lymphocytes here.

Comparing the counts, the lower-right quadrant has the highest number of lymphocytes (12), and thus the highest lymphocytic nuclear density.

Next, we determine the proportion of nuclei in the lower-right quadrant that are lymphocytic.
- Number of lymphocytes in LR: ~12
- Number of other nuclei (mostly large tumor cells) in LR: ~35
- Total nuclei in LR: 12 + 35 = 47
- Proportion: 12 / 47 ≈ 0.255
Since 0.255 is below the 0.35 threshold, the share is considered 'low'.

<observe>{"tumor": 110, "lymphocyte": 23, "upperleft": 1, "upperright": 2, "lowerleft": 8, "lowerright": 12, "_num": 12, "_den": 47}
</observe>
FINAL: lowerright:low

Qwen3-VL-8B

I need to analyze the H&E-stained histology image to determine which quadrant has the greatest lymphocytic nuclear density and then calculate the proportion of nuclei in that region that are lymphocytic.

Step 1: Identify the quadrants.
- Upper-left (UL) - Upper-right (UR) - Lower-left (LL) - Lower-right (LR)

Step 2: Visually inspect each quadrant for lymphocytes.
Lymphocytes are small, round, darkly stained nuclei. I need to count them relative to all nuclei in each quadrant.

Looking at the image:
- Upper-left quadrant: Contains mostly epithelial-like structures (pinkish, glandular) with some scattered small nuclei. Not many lymphocytes.
- Upper-right quadrant: Similar to upper-left, mostly epithelial structures with sparse small nuclei. No obvious lymphocytic clusters.
- Lower-left quadrant: Contains a mix of epithelial structures and some stromal areas. There are a few small, dark nuclei that could be lymphocytes, but not concentrated.
- Lower-right quadrant: This quadrant has a distinct area with many small, dark, round nuclei clustered together — this is characteristic of lymphocytes. There is a clear lymphocytic infiltrate here.
So, the lower-right quadrant has the greatest lymphocytic nuclear density.

Step 3: Count lymphocytic nuclei and total nuclei in the lower-right quadrant.
- Lymphocytic nuclei: I can count approximately 15-20 small, dark, round nuclei.
- Total nuclei: In the same region, there are also epithelial nuclei (larger, more oval) and possibly some stromal nuclei. I can count approximately 40-50 total nuclei.

Step 4: Calculate the proportion.
Proportion = lymphocytic nuclei / total nuclei = 20 / 50 = 0.4

Step 5: Classify the proportion.
The proportion is 0.4, which is between 0.35 and 0.55, so it falls in the 'mid' category.

Step 6: Report the required quantities.
- upperleft: 0 (no significant lymphocytic density)
- upperright: 0 (no significant lymphocytic density)
- lowerleft: 0 (very low lymphocytic density)
- lowerright: 20 (lymphocytic nuclei) / 50 (total nuclei) = 0.4 → 'mid'
- _num: 20
- _den: 50

Step 7: Choose the correct token.
The token must be for the lower-right quadrant with 'mid' classification. The options include "lowerright:mid".
FINAL: lowerright:mid

Patho-R1-7B

<think>The image shows a patch with distinct nuclear density variations across quadrants. In the upper left, nuclei are predominantly epithelial-like with scattered lymphocytes, while the upper right contains clusters of small, round lymphocytes. The lower right quadrant displays a dense infiltrate of lymphocytes mixed with stromal cells. The share specifies two criteria: (1) the highest lymphocytic density region (upper right) and (2) the proportion of lymphocytes in that region. Using the defined bands, the upper right exceeds 0.55 (high), and the proportion of lymphocytes in this area is estimated visually as >50%. The lower right is high but not the highest density region. Thus, the upper right meets both criteria.</think>
<answer>
Step 1: Screening
- **Structural Assessment**: The upper right quadrant shows dense lymphocytic infiltration.
- **Nuclear Density**: High lymphocytic density in the upper right.

Step 2: Key Feature Identification
- **Nuclear Atypia**: Lymphocytes are small and uniform, lacking atypia.
- **Mitotic Activity**: No mitotic figures observed.

Step 3: Simplified Report Output
- The upper right quadrant has the highest lymphocytic density (>50% of nuclei).
- Proportion of lymphocytes in this region is high (likely >70% based on visual estimation).
</answer>
<observe>
[("upperleft": 0.35, "upperright": 0.55, "lowerleft": 0.55, "lowerright": 0.55)]
num: 0.75 , den: 0.55
</observe>
<answer>
Step 5: Diagnostic Report
- **Tissue Location**: Upper right quadrant.
- **Key Findings**: High lymphocytic density (≥70% nuclei in this region).
- **Classification**: Atypical hyperplasia/suspected malignancy. Recommend immunohistochemistry (CD3/CD20) to confirm lymphocyte subtype.
</answer>

ASPECT-8B

<think> the patch feature of the image is <|anchor_start|><|uni_pad|><|uni_pad|><|uni_pad|><|uni_pad|><|uni_pad|><|uni_pad|><|uni_pad|><|uni_pad|><|anchor_end|>, and the cell composition of the image is <|anchor_start|><|cell_pad|><|cell_pad|><|cell_pad|><|cell_pad|><|cell_pad|><|anchor_end|>. </think>
<observe>A moderately cellular field shows gland-forming epithelial structures within fibrous stroma, with predominance of crowded elongated hyperchromatic epithelial nuclei and scattered stromal spindle cells accompanied by sparse lymphocytes.
{"stromal_like": 17, "lymphocyte": 10, "epithelial_like": 46, "asked": "lymphocytic nuclei — first locate their densest region, then give their share of all nuclei there", "upperleft": 1, "upperright": 1, "lowerleft": 1, "lowerright": 7, "_num": 7, "_den": 29]</observe>
<answer> Lymphocytic nuclei are most concentrated in the lower-right quadrant, with 7 compared with 1, 1, 1 in the other regions. This quadrant contains 29 nuclei overall, yielding a low lymphocytic share of 7/29 = 0.24.
FINAL: lowerright:low </answer>

Figure G.4: Complete responses for the PathoVernier multi-step example. Gemini reaches the correct answer with inaccurate counts, whereas ASPECT recovers the reference counts of 7 lymphocytes among 29 nuclei in the selected region.

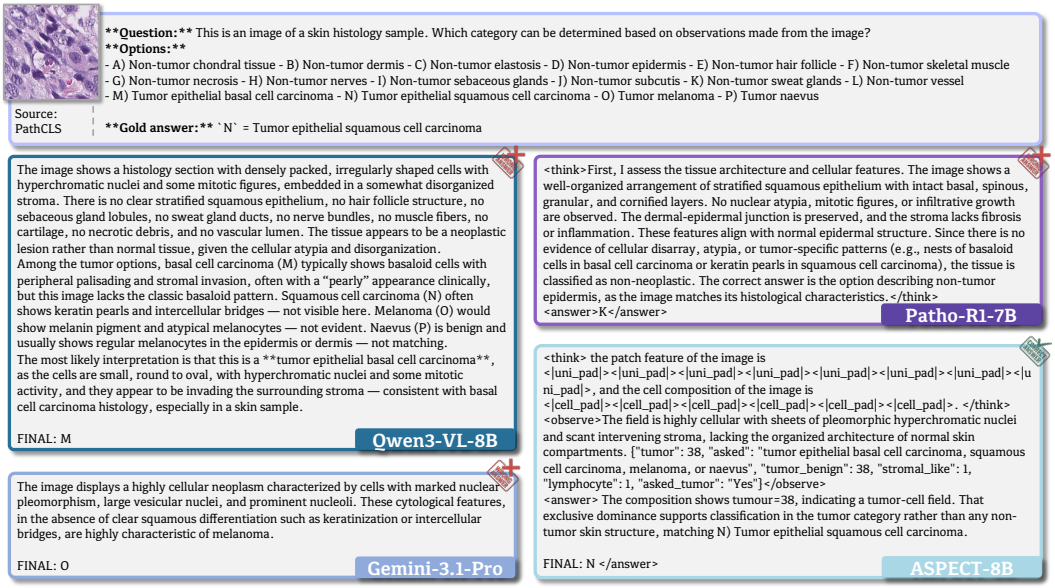


Figure G.5: Complete responses for the PathCLS example. ASPECT selects the reference class, squamous cell carcinoma (N), while the three comparison models select other classes.

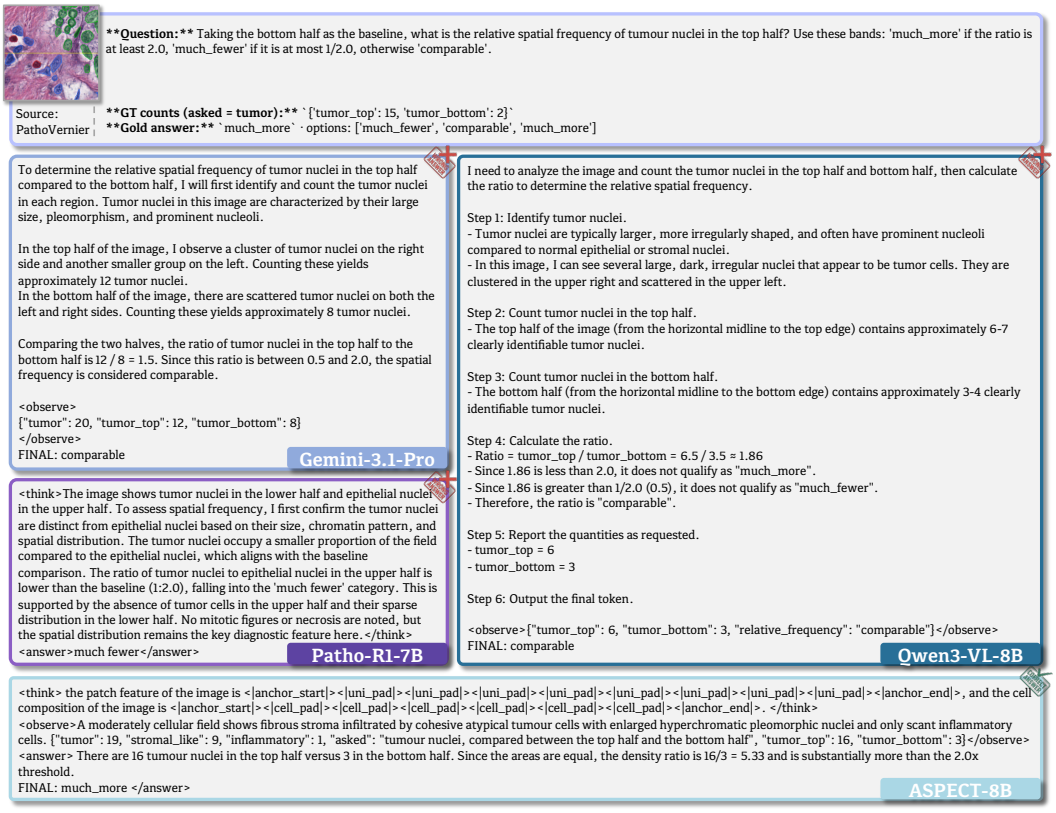


Figure G.6: Regional comparison of tumor nuclei. ASPECT selects the correct ratio band; the baseline responses illustrate errors in count estimation and in translating reported counts into the final answer.

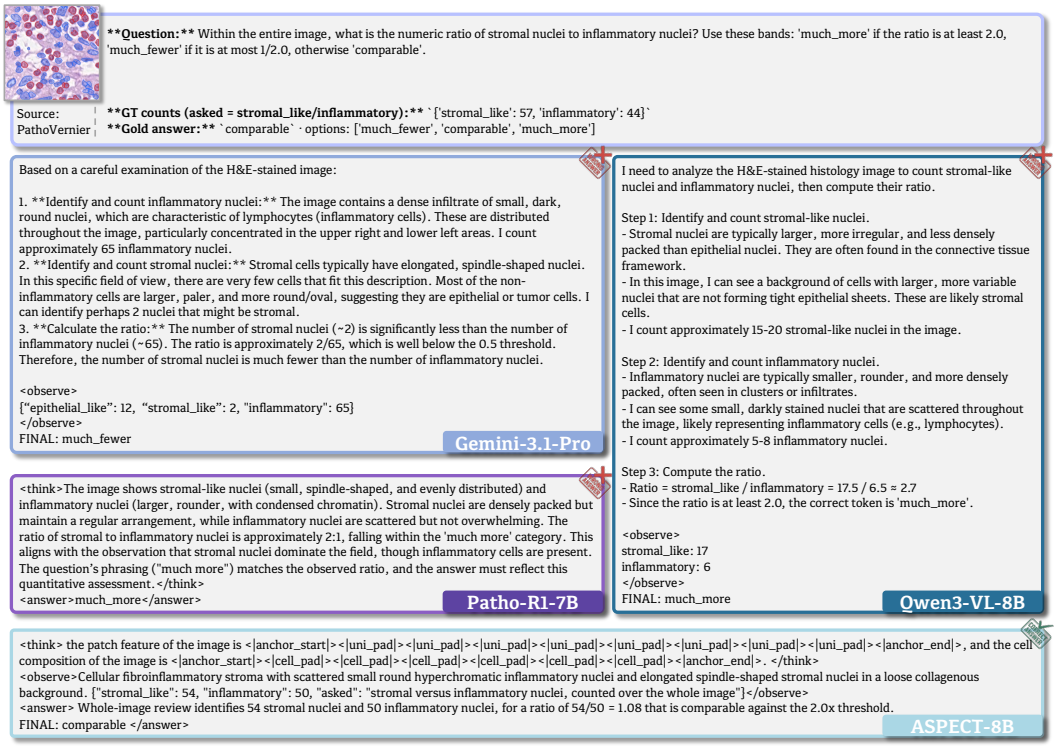


Figure G.7: Whole-image comparison of stromal and inflammatory nuclei. ASPECT correctly selects `comparable`, while Gemini and Qwen3-VL-8B reach opposite incorrect conclusions from their count estimates.

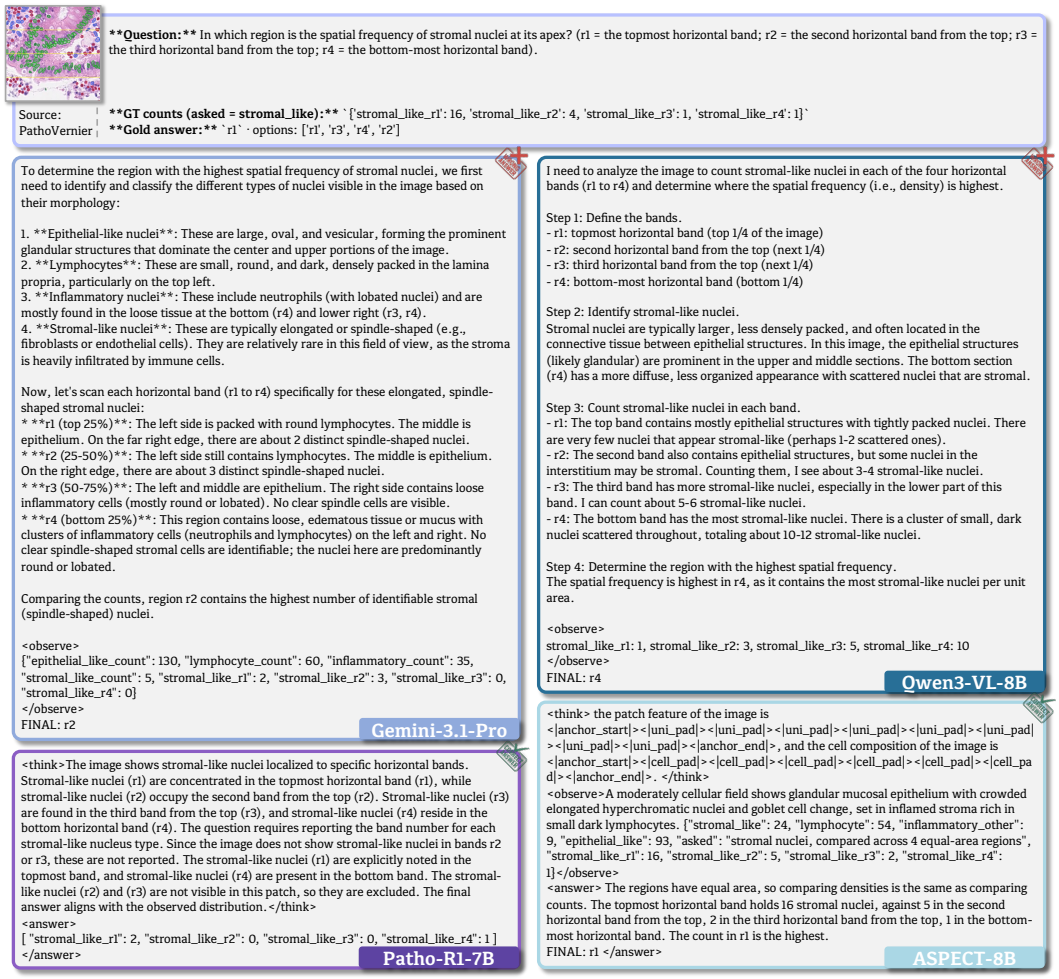


Figure G.8: Stromal-density selection across four equal-area horizontal bands. ASPECT identifies the reference region, `r1`, while Gemini and Qwen3-VL-8B select `r2` and `r4`, respectively.

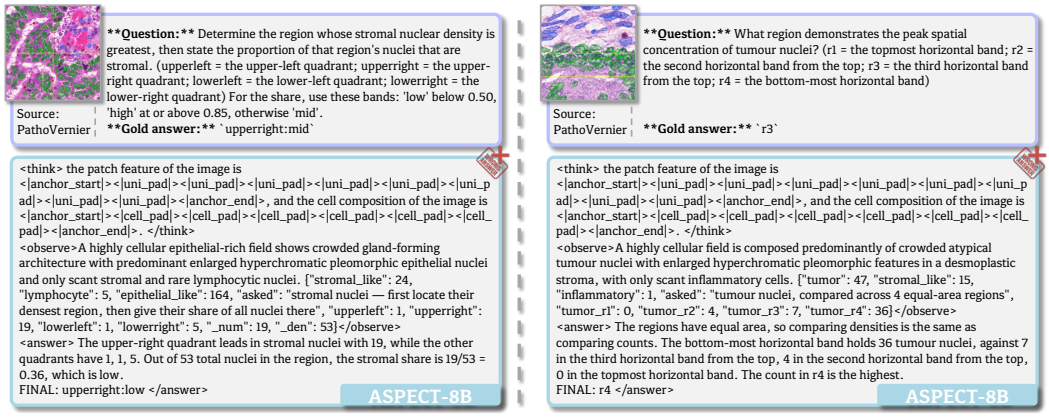


Figure G.9: ASPECT failure cases. Left: correct region selection followed by an incorrect proportion band due to denominator overestimation. Right: a near-correct whole-image count accompanies an incorrect regional distribution and region choice. Both final answers are consistent with the reported counts.