LEARNING WHEN TO RECONSIDER: ADEQUACY-GUIDED INFERENCE-TIME INTERVENTIONS FOR HISTOPATHOLOGY VLM AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Visual agentic systems that work with very large images often cannot process the whole image at once. Instead, they inspect selected regions (patches), ask a large vision–language model (LVLM) to describe what is visible, and use these descriptions to guide later navigation and reasoning. Inadequate or visually irrelevant descriptions can therefore affect the performance of the downstream task. Existing correction methods applied to LVLMs during test-time typically modify attention, decoding, or prompts to reduce visually unsupported responses, but each leaves the agentic system with a decision that cannot be resolved simply by using a stronger model: *when should a frozen LVLM perceptor be corrected, and which correction should be used?* We study this question in computational pathology, where agents navigate gigapixel scans of tissue specimens known as whole-slide images (WSIs) for visual question answering (VQA), using four LVLMs, three interventions, and two whole-slide image agents. Our experiments reveal two regularities. Descriptions judged relevant, sufficiently complete, and consistent with the ground truth evidence -which we call *adequate descriptions*-support higher downstream answer accuracy than descriptions judged inadequate. Secondly, intervention benefits depend on the model, question type, and baseline evidence quality of the initial description, and the same intervention can repair some descriptions while damaging others. To test whether these empirical patterns transfer to WSI VQA agentic systems and can support useful decisions by selecting one intervention family for low-confidence descriptions and accepting candidates only when their estimated adequacy increases by a fixed margin, we propose RESCUE–ROUTE–RESCORE, a simple controller. On a held-out patch-level benchmark, the proposed controller raises macro description adequacy from 55.7% to 67.2% and answer accuracy from 56.9% to 62.3%. Transferred to agentic systems that perform WSI level VQA without any WSI-level tuning, it achieves the highest point estimate of macro answer accuracy in all six dataset–agent comparisons. These findings suggest that intermediate evidence quality can serve as a transferable control signal and that visual agentic systems can improve by learning which evidence to reconsider. Code will be made available on acceptance.

1 INTRODUCTION

Agentic visual systems(Wu & Xie, 2024) are emerging as a powerful means of analyzing whole slide images (WSIs) in histopathology for various clinical and research applications, e.g., cancer diagnostics, treatment personalization and biomarker discovery (Truhn et al. (2026). These systems rarely call a large vision–language model (LVLM) only once (Chen et al. (2025a),Yang et al. (2026)). Instead, the LVLM acts as a "perceptor" within a longer loop: an agent selects a region of a WSI (i.e., an image patch), asks the model to describe it, stores the response, and uses the accumulated evidence to decide where to look next and/or how to answer the original question. This decomposition is particularly useful for gigapixel WSIs, which are too costly to process exhaustively at diagnostic resolution and consist mostly of irrelevant data (e.g., background or tissue lacking diagnostic value). However, this decomposition also creates a distinctive failure mode: a weak patch description is not merely a local error; once written to memory, it may alter later retrieval, zooming,

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.

and adjudication for WSI- or patient-level decisions. Existing test-time corrections offer several ways to alter an LVLM response to mitigate hallucinations and increase visual saliency, including visual-attention biasing, contrastive decoding, and prompted review (Liu et al., 2024; Sicong Leng, 2023; Lee et al., 2024; Liao et al., 2025). These methods are commonly evaluated applying the same intervention and configuration across inputs. Such a global rule optimizes average performance but cannot account for instance-level variation in whether an intervention will help. This may result in added computation for an already adequate description or replacing useful evidence with a poorer candidate.

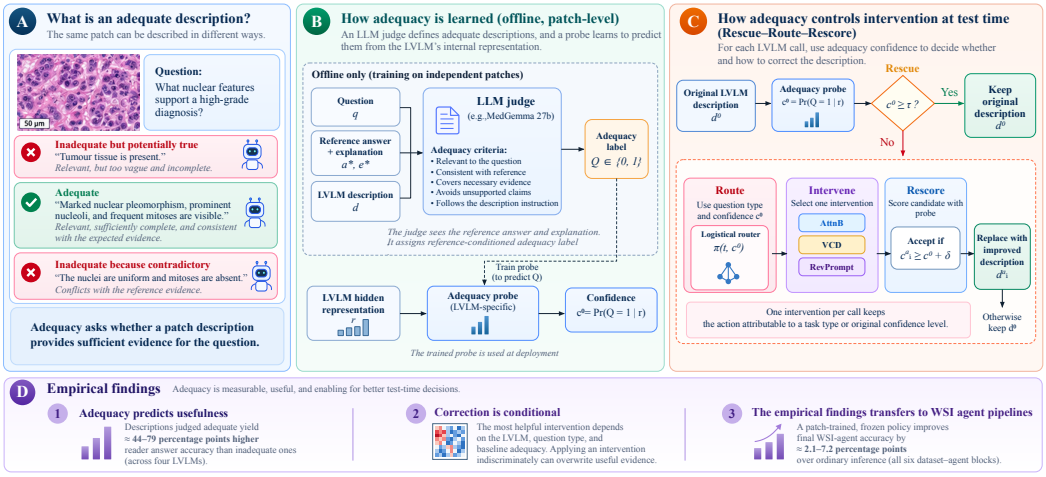


Figure 1: **Forming a control signal from intermediate visual evidence quality through *description adequacy*.** (a) A patch description is adequate when it communicates relevant, sufficiently complete, and reference-consistent evidence for the question. A description may be locally true yet inadequate because it omits the discriminating evidence. (b) During development, a medical expert LLM judge uses the question, reference answer, and explanation to label descriptions. These labels train a model-specific probe that estimates adequacy confidence from the LVLM's hidden representation. (c) At test time, high-confidence descriptions are retained. For a low-confidence description, the router uses question type and adequacy confidence to select one intervention family. The LVLM then re-describes the same patch using modified visual attention, contrastive decoding, or a review instruction. The candidate replaces the original only if its rescored adequacy confidence increases by at least the frozen margin δ. The resulting policy is transferred unchanged into WSI agents.

That leaves us a question faced by an agent at each perception turn: *should this particular description be corrected, and if so, which correction is appropriate?* The challenge is to find a local signal that is meaningful before the agent reaches a final answer. Patch-answer correctness is unsuitable for WSI navigation because one patch may contain evidence that is faithful but vague or conflicting with the WSI-level answer. Furthermore, optimizing patch-level answer correctness can be misleading due to reward ambiguity. Different descriptions can receive the same patch-level correctness reward despite providing very different amounts of useful evidence for downstream multi-patch decisions. Thus, we propose to use *adequacy* instead of correctness as the signal to guide multiple-step reasoning. The key distinction is between naming a correct finding and communicating the evidence needed for the question. Fig. 1A makes the distinction concrete. For a question about nuclear features, "Tumor tissue is present" may be correct but too vague to be useful (inadequate). Similarly, a description that fails to identify the relevant features or contradicts them is equally inadequate. That leads us to another question: *Can the quality of intermediate visual evidence support selective test-time control?* We therefore study *description adequacy*, in other words, whether a description given by the LVLM conveys the question-relevant evidence expected by a ground truth reference answer and explanation. This gives us an intermediate evaluation target before the agent has enough evidence to commit to a slide-level answer. We organize the study around three research questions. **RQ1:** Does the quality of the descriptive evidence (measured by description adequacy) track downstream usefulness, and can it be estimated without a reference explanation at test time? **RQ2:** How do

the model, the question type and baseline quality of the evidence in the initial LVLM description influence the effect of various interventions within each LVLM? **RQ3:** Do these relations transfer to WSI agentic pipelines well enough for an intervention routing policy trained on independent patches to improve downstream VQA performance, without training on WSIs or trajectories?

For **RQ1**, we find that on the PathMMU benchmark (Sun et al. (2024)), answerability rises sharply with description adequacy for every evaluated pathology LVLM (Fig. 2), suggesting that adequacy is both local and behaviorally relevant. Furthermore we find that adequacy alone does not determine how to correct a failure. For **RQ2**, we evaluate three parameter-free interventions acting at distinct control points within the LVLM: visual-token attention biasing (ATTNB; Liu et al., 2024), visual contrastive decoding (VCD; Sicong Leng, 2023), and review prompting (REVPROMPT; Lee et al., 2024; Liao et al., 2025). Paired treatment analyses reveal the paper's second main empirical insight: **intervention utility is conditional** on task type and baseline evidence quality. The preferred action changes across LVLMs and question types, and its repair–damage balance can change again with the baseline adequacy confidence of the ordinary description as shown in Fig. 3. For studying **RQ2** and **RQ3**, and to test whether these patterns can guide decisions, we propose RESCUE–ROUTE–RESCORE (Fig. 1C). A small probe first estimates whether the original description is adequate. If the description adequacy confidence is low, a router uses the question type and that confidence to choose a correction. The LVLM then describes the same patch again, and the original is replaced only if the new description receives a sufficiently higher probe score (i.e., higher adequacy confidence).

Our contributions are empirical and methodological: (i) we establish description adequacy as a predictable, behaviorally relevant intermediate target; (ii) we identify model–task–quality dependence in three test-time interventions and show that an auditable low-capacity policy captures the useful empirical structure; and (iii) we demonstrate that this patch-level low-capacity policy of correction transfers to two multi-step WSI agents without access to WSI labels or trajectories during fitting.

2 RELATED WORK

Pathology models and whole slide understanding. Domain-specific pathology LVLMs have been developed through complementary forms of specialization. Quilt-LLaVA uses pathology-specific visual instruction tuning derived from localized educational narratives (Seyfioglu et al., 2024), while Patho-R1 adds reinforcement-learning-based pathology reasoning (Zhang et al., 2026b). MedGemma 1.5 extends a general medical foundation model with improved multimodal and histopathology capabilities (Sellergren et al., 2026). PathMMU provides expert-level pathology questions together with reference answers and explanations, making it suitable for studying both answer correctness and the quality of intermediate descriptions (Sun et al., 2024). Importantly, recent evidence indicates that higher pathology benchmark accuracy does not necessarily imply stronger image–text grounding: models can exploit textual priors while attending only weakly to diagnostic visual evidence (Zhang et al., 2026a). This motivates our separation of *local description adequacy* from final-answer accuracy. Unlike work that trains or evaluates a new pathology LVLM, we freeze each LVLM and study how its evidence can be selectively improved at test time. At the WSI level, WSI-VQA introduced generative whole-slide question answering (Chen et al., 2024). SlideChat provides a WSI assistant and evaluation benchmarks (Chen et al., 2025b), including a breast cancer needle biopsy (BCNB) arm based on the cohort of Xu et al. (2021). PathReasoner-R1 learns structured pathology reasoning (Jiang et al., 2026), while PathAgent and PathNavigate combine patch acquisition with evidence aggregation (Chen et al., 2025a; Yang et al., 2026). We study whether correcting their local descriptions improves complete WSI answers without fitting on slides or trajectories.

Test-time correction. PAI strengthens visual influence (Liu et al., 2024), and visual contrastive decoding compares predictions from original and perturbed images (Sicong Leng, 2023). VOLCANO learns feedback-guided revision (Lee et al., 2024), while Liao et al. (2025) examine correction of semantic grounding errors. Building on these mechanisms, we investigate how their benefits vary and whether description adequacy can guide when to intervene, which intervention to attempt, and whether to accept its output.

3 Problem Formulation

Represent WSI S_n as a multiresolution pyramid $\mathcal{P}_n = \{S_n^{(\ell)}\}_{\ell=1}^{L}$, where each level shows the same tissue at a different resolution. Level ℓ contains sub-regions or patches $\{I_{n,\ell,j}\}_{j=1}^{J_{n,\ell}}$, with j indexing a spatial tile. Thus (ℓ, j) identifies its resolution and location. The agentic systems in this study select patches via their acquisition policies and supply patch descriptions to a downstream reasoner.

Let i index a patch–question instance (see Appendix A), either from an independent image or a selected WSI region. The frozen LVLM f_m generates the ordinary description

$$d_i^0 = f_m(I_i, q_i). \tag{1}$$

Our controller decides whether to retain or replace this description before downstream use. Intervention therefore operates locally, while WSI answer correctness is evaluated for the complete agent run. We define *description adequacy* as conveying question-relevant, sufficiently complete, and reference-consistent evidence. An offline text judge assigns $Q_i^0 \in \{0, 1\}$ from the description, question, and reference answer and explanation (Fig. 1A–B). At deployment, a probe estimates the probability of adequacy, c_i^0, from the LVLM's hidden representation without requiring the judge or reference. We use this estimate to guide selective correction. Unless stated otherwise, "confidence" refers to this probe-estimated probability, whereas Q_i^0 denotes the judge's binary label. We denote the sets of LVLMs, interventions and question types by $\mathcal{M}$, $\mathcal{A}$, and $\mathcal{T}$, respectively.

4 Empirical Motivation for Method

We examine whether adequacy tracks downstream usefulness and whether intervention benefits vary systematically. Section 6 provides the evaluation protocol.

4.1 Adequacy tracks downstream usefulness

We compare judge-assigned adequacy with the accuracy of a separate downstream LLM reader that answers using only the question, options, and LVLM description. The reader sees neither the image nor the reference answer. This tests whether adequacy tracks downstream usefulness without defining adequacy by the reader's correctness. Across all four LVLMs, descriptions with higher adequacy score from the judge LLM are markedly more answerable, with accuracy rising most sharply in higher-quality bins (Fig. 2). This association is not causal and does not make adequacy a substitute for answer confidence. It does, however, show that the local adequacy label tracks evidence that matters to a downstream reader.

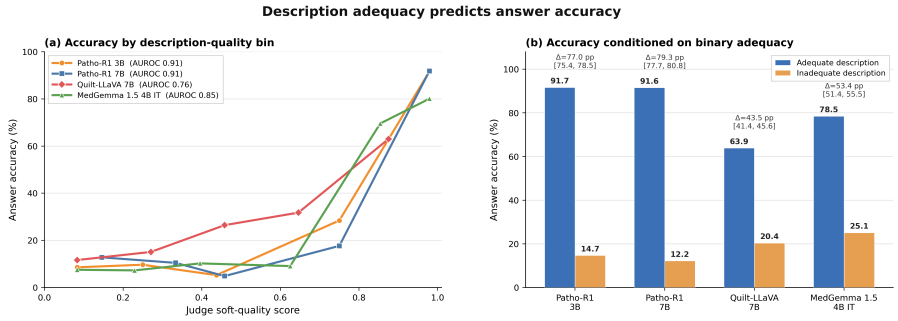


Figure 2: **Adequacy is behaviorally relevant across LVLMs.** Left: Answer accuracy rises with the frozen judge's continuous description score. The item-level AUROC measures using this score to discriminate correct from incorrect reader LLM answers (see legend). Right: Descriptions meeting the binary adequacy criterion are more answerable for all four models. The judge uses reference answers and explanations, whereas the answerer sees only the question, options, and description.

4.2 INTERVENTION BENEFIT DEPENDS ON TASK AND BASELINE CONFIDENCE

We compare 3 interventions on the same image-question instances: ATTNB increases attention to image tokens, VCD contrasts predictions from original and blurred images, and REVPROMPT supplies a neutral review instruction (Appendix A.4). Each candidate is compared with the same LVLM's ordinary description. The plotted policies intervene only when $c_i^0 < 0.60$, using a threshold selected on pooled validation adequacy. These analyses motivate the role of the controller.

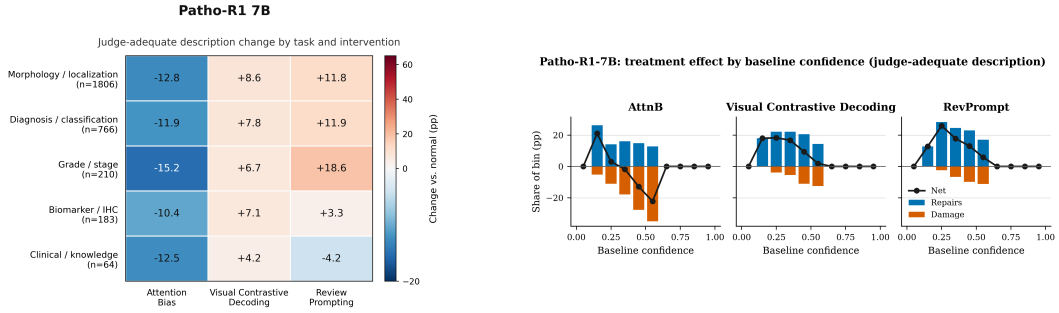


Figure 3: **Intervention utility is conditional, illustrated for Patho-R1 7B.** The baseline confidence is the baseline description adequacy confidence from the probe for the initial LVLM response. Left: Change in description adequacy relative to ordinary inference by question type. Right: The y axis shows % of descriptions in that particular confidence bin that were repaired/harmed. Repair rate, damage rate, and net effect vary as a function of the original description's confidence bin, and task-level averages can hide useful low-confidence regions. Fig. 9 and 10 provide results for all four LVLMs. Descriptions with $c_i^0 \geq 0.60$ are retained by the gate, giving zero change by construction.

In Fig. 3 we can see an example for Patho-R1 7B. Two clear patterns emerge. First, an intervention's value depends on the question type. For Patho-R1 7B, review prompting improves adequacy by 11.8 percentage points (pp) on morphology/localization questions but reduces it by 4.2 pp on clinical/knowledge questions. The same correction can therefore be helpful in one task and harmful in another. Second, question type is not the whole story. The confidence-stratified results show that attention biasing repairs many very-low-confidence descriptions, despite having a negative task-averaged effect, but becomes increasingly damaging outside that region. The complete results confirm that both forms of heterogeneity also vary considerably across LVLMs (Figures 9 and 10). These findings suggest that task type correlates with *what evidence is being sought*, while baseline confidence helps indicate *whether the current description is likely to benefit from correction*. A universal intervention ignores both signals, whereas task-only selection ignores differences within a task. In the next phase we therefore test whether combining question type and ordinary-description confidence is sufficient to support useful routing within each LVLM.

5 METHOD: RESCUE–ROUTE–RESCORE

RESCUE–ROUTE–RESCORE uses estimated adequacy to decide whether to intervene, which intervention to attempt, and whether to accept its candidate (Fig. 1C). See Fig. 5 for detailed workflow.

When to intervene. Offline fitting proceeds in two stages. First, for each LVLM m, we train a one-hidden-layer adequacy probe g_m. Given the hidden representation r_i^0 of the ordinary description produced by the LVLM m, the probe and its Platt calibrator produce

$$c_i^0 = \mathrm{Cal}_m\big(g_m(r_i^0)\big) \approx \Pr(Q_i^0 = 1 \mid r_i^0), \tag{2}$$

Here $Q_i^0 = 1$ means that the offline judge labeled the ordinary description as adequate. The probe estimates the quality of the description already produced. We then fit the intervention router using these calibrated confidences and question types, as described below. The probe, calibrator, and router are frozen at test time. If $c_i^0 \geq \tau$, the gate retains d_i^0. Otherwise, it invokes the router. Thus, the probe asks *does the ordinary description need help?*, while the router asks *which intervention is most likely to help?* Neither the adequacy judge nor the reference answer is required.

Which intervention to use. The fixed action set $\mathcal{A} = \{\text{ATTNB}, \text{VCD}, \text{REVPROMPT}\}$ spans three different control points while preserving the LVLM and the agent's text interface. ATTNB reweighs visual attention (Liu et al., 2024), whereas VCD contrasts original- and blurred-image logits (Sicong Leng, 2023). REVPROMPT regenerates under a neutral pathology-review procedure proposed by a frozen text model (Lee et al., 2024; Liao et al., 2025). A frozen question-only classifier assigns one of five coarse types to each question: morphology/localization, diagnosis/classification, grade/stage, biomarker/IHC, or clinical/knowledge. It never sees the image, description, candidate, or outcome. We select one intervention family before generation to test whether task and adequacy confidence can guide that choice. Combining mechanisms would require estimating their interaction effects, while generating every family would add computation and a further selection stage. Our restriction keeps the selected family and its cost inspectable (Appendix A.4).

Let $\mathcal{T}$ contain the five question types listed above, and let $\mathbf{e}_{\mathcal{T}}(t_i)$ be the one-hot encoding of the predicted type. For a low-confidence instance i, let t_i be its predicted question type and let $\mathbf{e}_{\mathcal{T}}(t_i) \in \{0,1\}^{|\mathcal{T}|}$ be its one-hot encoding. Then as features, the router uses

$$x_i = \left[1,\ c_i^0,\ \mathbf{e}_{\mathcal{T}}(t_i)^\top,\ c_i^0 \mathbf{e}_{\mathcal{T}}(t_i)^\top\right]^\top \in \mathbb{R}^{2|\mathcal{T}|+2}. \tag{3}$$

The intercept, confidence, task indicators, and interactions give $2|\mathcal{T}| + 2 = 12$ features for five question types, independently of the number of LVLMs or interventions. The interactions allow confidence to have a different effect with each question type. During offline caching, intervention a produces candidate d_i^a, which the judge labels $Q_i^a \in \{0,1\}$. For each LVLM-intervention pair, logistic regression with coefficients $w_{m,a} \in \mathbb{R}^{2|\mathcal{T}|+2}$ estimates

$$u_{i,a} = \sigma\big(w_{m,a}^\top x_i\big) \approx \Pr(Q_i^a = 1 \mid c_i^0, t_i). \tag{4}$$

Here $u_{i,a}$ predicts whether intervention a will produce an adequate candidate, whereas c_i^0 estimates whether the existing description is adequate. The router then selects the intervention to be $a_i^* = \arg\max_{a \in \mathcal{A}} u_{i,a}$. For the adequacy-supervised controller, neither answer correctness nor WSI outcomes are training targets. The low capacity is deliberate: for each question type, intervention logits are linear in confidence, giving inspectable action boundaries (Fig. 4). This tests whether task and adequacy confidence suffice for useful selection. Appendix A.6 gives the fitting details. After selecting a_i^*, we apply that intervention to the frozen LVLM and ask it to describe the same image patch I_i for the same question q_i again. The intervention modifies attention, decoding, or the review instruction, producing a candidate description $d_i^{a_i^*}$ without updating the LVLM weights.

Whether to trust the candidate. The router predicts which intervention is promising, but its candidate may still be inadequate. We therefore pass the generated candidate back through the frozen LVLM as a fixed token sequence, without generating another response, and apply the same adequacy probe to its hidden representation. Attention biasing and contrastive decoding are disabled during this pass, while review prompting retains its revised instruction. We replace d_i^0 with the candidate $d_i^{a_i^*}$ only if

$$c_i^{a_i^*} \geq c_i^0 + \delta, \tag{5}$$

Otherwise, the original is retained. The validation-selected margin δ is frozen. Appendix A.7 gives further details. Probe fitting uses varied description prompts to reduce sensitivity to prompt wording, although robustness to the review instruction is approximate (Appendix A.2). No reference, correctness signal, trajectory outcome, or WSI supervision is available at deployment.

6 EXPERIMENTAL SETUP

Models and patch-level evaluation. We evaluate Patho-R1 3B & 7B, Quilt-LLaVA v1.5 7B, and MedGemma 1.5 4B, with all LVLM weights frozen. On the patch level PathMMU dataset, each model receives the image, question, and answer options and describes the relevant visual evidence without selecting an option. Each ordinary description is paired with candidates from ATTNB, VCD, and REVPROMPT generated for the same image–question instance. Detailed model configurations and intervention settings are given in Tab. 4 and Appendix A.4. For generating adequacy labels, a MedGemma-27B based LLM judge receives the question, options, reference answer and

explanation, and LVLM description, but not the image. It assesses relevance, reference consistency, coverage, unsupported claims, instruction following, and overall quality. Equation 10 specifies the binary adequacy criterion. For patch level downstream answering tests, a separate Qwen3.5-4B LLM reader answers from the question, options, and description without seeing the image/reference. This separates the supervision target from the downstream answer-accuracy measurement.

Fitting, evaluation splits and transfer to WSI agents. Probe and router fitting use development data, while policy selection uses a separate validation partition. Every development confidence used to fit the router is generated out of fold: the probe and calibrator scoring an image were fitted without any question or prompt variant derived from that image. The grouped fitting and calibration procedure is described in Appendix A.2. Final patch-level evaluation uses a held-out PathMMU split, with the probe, calibrator, router, gate, and acceptance rule frozen. Tab. 5 summarizes the roles of the data partitions. PathMMU based patch-level policies are evaluated using the same cached descriptions and intervention candidates. We compare ordinary inference, one validation-selected best fixed intervention, task-only lookup, confidence-only routing, the full router, and ROUTER+ACCEPT. Tab. 6 defines the policies. We additionally fit a matched controller using PathMMU answer correctness instead of description adequacy as its training target for WSI agent transfer comparison experiments. In this variant, both the confidence probe and the per-intervention logistic models predict whether the frozen reader answers correctly from the description. The two controllers use the same LVLM outputs, features, grouped data splits, model capacity, and validation procedure. Neither controller uses WSI labels or trajectories. After patch-level fitting, every controller component is frozen and applied at each PATHAGENT and SURPRISENAV perceptor call. Fig. 6 shows the integration. We evaluate all four LVLMs on SlideBench-TCGA, SlideBench-BCNB, and WSI-VQA without changing the agent logic or the maximum budget of seven ordinary perceptor calls per question. Owing to limited compute, *we restricted the LVLM call budget*; so, the ordinary agent baselines reported here are lower than the originally published results(see Appendix I). Tab. 7 gives the agent configurations. For WSI transfer, we also report a post-hoc "strongest fixed" baseline. For each LVLM-dataset-agent combination, it takes the highest observed test accuracy among the three single-intervention policies before averaging over LVLMs. It is therefore distinct from the validation-selected best fixed policy used in the PathMMU comparison.

Evaluation measures. We report judge-labeled description adequacy and reader answer accuracy for patch-level evaluation, and final-answer accuracy for complete WSI trajectories. Macro results average over the four LVLMs. Probe evaluation experiments report discrimination and calibration, while Tab. 17 and 18 report realized policy utilization and computational cost.

7 RESULTS

7.1 REFERENCE-FREE ADEQUACY ESTIMATION

On the held-out PathMMU evaluation, the model-specific probes achieve AUROC 0.678–0.779 and expected calibration error (ECE) 0.025–0.044 (Tab. 13 and Fig. 7). The separate signal comparison in Tab. 8 also favors absolute adequacy confidence over the other screened measures. This shows that the LVLM's internal representation contains information about description adequacy that is available without the judge or reference at deployment. These probes are informative but imperfect, which motivates both selective invocation and a subsequent check of the generated candidate.

7.2 PATCH-LEVEL ROUTING ON PATHMMU

Tab. 1 compares the policies across the four LVLMs. A fixed intervention improves macro adequacy, and conditioning on both task and confidence gives a further gain. Rescoring improves the routed result for every LVLM, as shown by the per-model results in Tab. 14 and Fig. 8. The macro averages in Tab. 1 quantify the gain across the models. Relative to ordinary inference, ROUTER+ACCEPT improves adequacy by 11.4 pp and answer accuracy by 5.4 pp. The corresponding gains over best fixed are 4.5 and 3.3 pp. Its advantage over both partial routers supports using task and confidence together on held-out examples. Moreover, adding rescoring based acceptance gating to the router contributes 1.4 adequacy pp and 1.1 accuracy pp in the same table. Choosing an intervention and deciding whether to trust its particular output are therefore useful, distinct decisions.

Table 1: Macro-average PathMMU results across the four LVLMs (%). Router+accept applies rescoring-based acceptance to the routed candidate. Per-model results are reported in Tab. 14.

Outcome	Ordinary	Best fixed	Conf.-only	Task lookup	Router	Router+accept
Description adequacy	55.7	62.7	64.4	63.8	65.8	**67.2**
Answer accuracy	56.9	59.0	59.3	59.6	61.2	**62.3**

7.3 Interpreting the fitted policy

The router's restricted form allows us to inspect the selected intervention across question types and baseline confidence. Fig. 4 visualizes the differences between the LVLMs. Appendix E.3 derives the boundaries. For each LVLM and question type, the router compares the predicted candidate adequacy probabilities of all three interventions and selects the largest. Because the sigmoid preserves ordering, action changes occur where the highest-scoring logit changes from one intervention to another. These task-specific boundaries differ from the shared gate $\tau = 0.60$, which determines whether any intervention is attempted. For Patho-R1 3B, RevPrompt occupies most of the displayed region. By contrast, Patho-R1 7B selects AttnB in four tasks below boundaries near 0.17–0.23 before switching to another arm (reflecting the pattern seen in Fig. 3). Quilt-LLaVA has several within-task switches, whereas MedGemma is mostly task-driven apart from a biomarker/IHC switch near 0.52. Note that these are fitted decision boundaries, not observed action frequencies, and regions to the right of $\tau = 0.60$ are never executed.

7.4 Frozen transfer to WSI agents

We next test whether the patch-trained controller improves complete WSI agents without further fitting. Tab. 2 compares frozen router+accept with ordinary inference and the post-hoc strongest fixed comparator defined in Section 6.

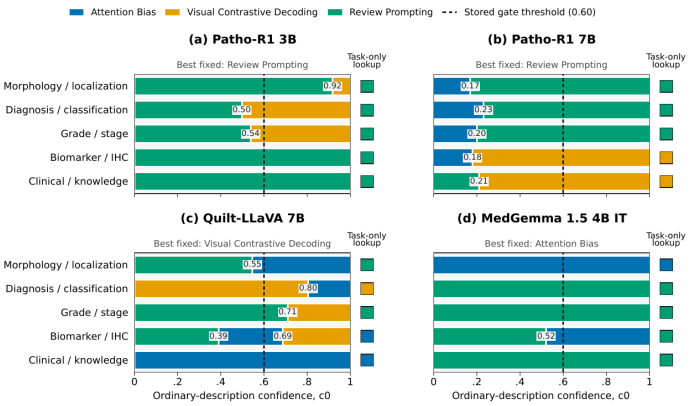


Figure 4: **The fitted router has explicit policy regions.** For each LVLM and question type, Equation 4 predicts the probability that each intervention will produce an adequate description. Color shows the intervention with the highest predicted probability as baseline adequacy confidence c^0 varies. Labeled boundaries mark where the preferred intervention changes because two competing predictions cross (Appendix E.3). The dashed line is the separate intervention gate: only cases with $c^0 < 0.60$ are routed. Squares show task-only choices, and panel text identifies the best fixed intervention. These regions describe action selection before candidate generation and acceptance.

Router+Accept leads all six macro blocks in Tab. 2, gaining 2.07–7.17 pp over ordinary inference and 0.66–1.67 pp over strongest fixed. The full matrices in Tab. 9–11 show that it is best in $23/24$ individual combinations. The exception is Patho-R1 3B with SurpriseNav on TCGA, where it equals Only-VCD (Tab. 9). On WSI-VQA, it improves all eight model–agent combinations even though several single-arm policies regress (Tab. 11). Together, this shows that a local target learned without slides or trajectories can guide correction in both agent loops. A matched controller trained on patch-level answer correctness instead of adequacy underperforms on all six

Table 2: Frozen-policy transfer, macro-averaged over the four LVLMs (accuracy, %). "Strongest fixed" is the per-LVLM maximum among the three fixed-action test results, serving as a conservative post-hoc comparator. Δ_0 and Δ_{fix} are Router+Accept gains over ordinary and strongest fixed inference.

Dataset	Pipeline	Ordinary	Strongest fixed	ROUTER+ACCEPT	Δ_0	Δ_{fix}
SlideBench-TCGA	PATHAGENT	38.44	43.94	**45.61**	+7.17	+1.67
SlideBench-TCGA	SURPRISENAV	38.51	41.19	**42.16**	+3.65	+0.97
SlideBench-BCNB	PATHAGENT	41.77	43.91	**45.16**	+3.39	+1.25
SlideBench-BCNB	SURPRISENAV	40.04	42.02	**42.68**	+2.63	+0.66
WSI-VQA	PATHAGENT	29.99	30.92	**32.06**	+2.07	+1.14
WSI-VQA	SURPRISENAV	25.93	26.71	**28.03**	+2.11	+1.32

transfer settings (mean +1.48 vs +3.5 pp, Tab. 12), supporting adequacy as the better supervision target. We see that correctness supervision still improves over ordinary inference in every setting, showing that patch-level answer correctness contains a useful control signal. Fig. 4 describes which action the router would choose, but it does not show how often those situations occur. Table 17 provides that missing deployment view. Across the eight agent-LVLM settings, the gate activates on 35.4–72.2% of perceptor calls and ultimately replaces 19.5–51.1% of descriptions; the realized action shares also show that all three interventions are used. This selectivity adds 17–67 s of median latency and 0.47–2.00 GPU-hrs per 100 questions, corresponding to 4.7–26.8 incremental GPU-minutes per wrong-to-right repair (Tab. 18).

8 DISCUSSION AND LIMITATIONS

Our experiments show that description adequacy tracks downstream usefulness and the conditional nature of interventions. A fundamental constraint of the proposed approach is that better patch descriptions do not guarantee a correct slide-level answer, because navigation, evidence aggregation and downstream reasoning can still fail. Despite this, when transferred to agentic systems that perform WSI-level VQA without any WSI-level tuning, the proposed RESCUE–ROUTE–RESCORE method achieved the highest point estimate of macro answer accuracy in all six dataset–agent comparisons, demonstrating the feasibility of using patch-level description adequacy as a control signal for WSI-level agents. The study has some specific limitations. First, the simple linear router policy may not be optimal for selecting the best intervention. However, this design was deliberately chosen to prioritize policy-level interpretability, as it provides inspectable decision regions. Second, the reported wrong-to-right changes are paired at the question level and attribute improvement to the complete policy, not to a particular repaired patch description. Because corrected evidence can affect later aggregation, reasoning, and, in PathAgent, subsequent acquisition decisions, we cannot establish that the specific patches whose descriptions were replaced caused the final answer to change. Third, adequacy is assigned by a reference-conditioned text judge rather than through direct image inspection by pathologists. It measures relevance, completeness and consistency with the reference, but not image-grounded factuality. Finally, the scope of our conclusions remains limited to four pathology LVLMs, three intervention families and two agents. Further limitations and implementation-specific threats are discussed in Appendix G and I.

9 CONCLUSION

Here, we show that description adequacy can serve as a test-time control signal for visual agents in histopathology. Adequate descriptions support higher downstream answer accuracy, and adequacy can be estimated reference-free at deployment time. However, test-time interventions are not uniformly beneficial: their effect depends on the LVLM, the question type, and the quality of the initial description. This heterogeneity motivates selective control over uniform application. A simple controller built on these regularities improves description adequacy at the patch level. Without any WSI-level tuning, it transfers to frozen whole-slide agents, where it achieves the highest accuracy point estimates across all dataset–agent comparisons. Evaluating evidence quality as a control signal in other large-image domains is a natural next step.

AI USE STATEMENT

We used generative AI tools to edit, correct and compact manuscript text and figure captions. The authors reviewed the AI-assisted material and take responsibility for the final methods, citations, results, and claims.

REPRODUCIBILITY STATEMENT

The main text specifies the controller and evaluation protocol. Appendix A details candidate generation, probe fitting, routing, and acceptance; Appendix B records data splits, model configurations, and agent settings; and Appendices F and I document frozen choices and compute. Code will be made available upon acceptance, as stated in the abstract.

REFERENCES

Anwoy Chatterjee, Yash Goel, and Tanmoy Chakraborty. HIDE and seek: Detecting hallucinations in language models via decoupled representations. *arXiv preprint arXiv:2506.17748*, 2025. URL https://arxiv.org/abs/2506.17748.

Jingyun Chen, Linghan Cai, Zhikang Wang, Yi Huang, Songhan Jiang, Shenjin Huang, Hongpeng Wang, and Yongbing Zhang. PathAgent: Toward interpretable analysis of whole-slide pathology images via large language model-based agentic reasoning. *arXiv preprint arXiv:2511.17052*, 2025a. URL https://arxiv.org/abs/2511.17052.

Pingyi Chen, Chenglu Zhu, Sunyi Zheng, Honglin Li, and Lin Yang. WSI-VQA: Interpreting whole slide images by generative visual question answering. In *Computer Vision – ECCV 2024*, pp. 401–417. Springer, 2024. doi: 10.1007/978-3-031-72764-1_23.

Ying Chen, Guoan Wang, Yuanfeng Ji, Yanjun Li, Jin Ye, Tianbin Li, Ming Hu, Rongshan Yu, Yu Qiao, and Junjun He. SlideChat: A large vision-language assistant for whole-slide pathology image understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5134–5143, 2025b. doi: 10.1109/CVPR52734.2025.00484.

Nhat Hoang, Minh Vu, My T. Thai, and Manish Bhattarai. PAS: Prelim attention score for detecting object hallucinations in large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18273–18283, 2026.

Zhi Huang, Federico Bianchi, Mert Yuksekgonul, Thomas J. Montine, and James Zou. A visual–language foundation model for pathology image analysis using medical twitter. *Nature Medicine*, 29(9):2307–2316, 2023. doi: 10.1038/s41591-023-02504-3.

Songhan Jiang, Fengchun Liu, Ziyue Wang, Linghan Cai, and Yongbing Zhang. PathReasoner-R1: Instilling structured reasoning into pathology vision-language models via knowledge-guided policy optimization. *arXiv preprint arXiv:2601.21617*, 2026. URL https://arxiv.org/abs/2601.21617.

Zhangqi Jiang, Junkai Chen, Beier Zhu, Tingjin Luo, Yankun Shen, and Xu Yang. Devils in middle layers of large vision-language models: Interpreting, detecting and mitigating object hallucinations via attention lens. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 25004–25014, 2025.

Seongyun Lee, {Sue Hyun} Park, Yongrae Jo, and Minjoon Seo. Volcano: Mitigating multimodal hallucination through self-feedback guided revision. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Long Papers*, Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2024, pp. 391–404. Association for Computational Linguistics (ACL), 2024. doi: 10.18653/v1/2024.naacl-long.23. Publisher Copyright: ©2024 Association for Computational Linguistics.; 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2024 ; Conference date: 16-06-2024 Through 21-06-2024.

Yuan-Hong Liao, Rafid Mahmood, Sanja Fidler, and David Acuna. Can large vision-language models correct semantic grounding errors by themselves? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14667–14678, 2025.

Shi Liu, Kecheng Zheng, and Wei Chen. Paying more attention to image: A training-free method for alleviating hallucination in lvlms. *arXiv preprint arXiv:2407.21771*, 2024.

Ming Y. Lu, Bowen Chen, Drew F. K. Williamson, Richard J. Chen, Ivy Liang, Tong Ding, Guillaume Jaume, Igor Odintsov, Long Phi Le, Georg Gerber, Anil V. Parwani, Andrew Zhang, and Faisal Mahmood. A visual-language foundation model for computational pathology. *Nature Medicine*, 30(3):863–874, 2024. doi: 10.1038/s41591-024-02856-4.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Andrew Sellergren, Chufan Gao, Fereshteh Mahvar, Timo Kohlberger, Fayaz Jamil, Madeleine Traverse, Alberto Tono, Bashir Sadjad, Lin Yang, Charles Lau, Liron Yatziv, Tiffany Chen, Bram Sterling, Kenneth Philbrick, Richa Tiwari, Yun Liu, et al. MedGemma 1.5 technical report. *arXiv preprint arXiv:2604.05081*, 2026. URL https://arxiv.org/abs/2604.05081.

Mehmet Saygin Seyfioglu, Wisdom O. Ikezogwo, Fatemeh Ghezloo, Ranjay Krishna, and Linda G. 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, 2024.

Guanzheng Chen Xin Li Shijian Lu Chunyan Miao Lidong Bing Sicong Leng, Hang Zhang. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. *arXiv preprint arXiv:2311.16922*, 2023. URL https://arxiv.org/abs/2311.16922.

Yuxuan Sun, Hao Wu, Chenglu Zhu, Sunyi Zheng, Qizi Chen, Kai Zhang, Yunlong Zhang, Dan Wan, Xiaoxiao Lan, Mengyue Zheng, Jingxiong Li, Xinheng Lyu, Tao Lin, and Lin Yang. PathMMU: A massive multimodal expert-level benchmark for understanding and reasoning in pathology. In *Computer Vision – ECCV 2024*, pp. 56–73. Springer, 2024. doi: 10.1007/978-3-031-73033-7_4.

The Cancer Genome Atlas Research Network, John N. Weinstein, Eric A. Collisson, Gordon B. Mills, Kenna R. Mills Shaw, Brad A. Ozenberger, Kyle Ellrott, Ilya Shmulevich, Chris Sander, and Joshua M. Stuart. The cancer genome atlas pan-cancer analysis project. *Nature Genetics*, 45 (10):1113–1120, 2013. doi: 10.1038/ng.2764.

Daniel Truhn, Shekoofeh Azizi, James Zou, Leonor Cerda-Alberich, Faisal Mahmood, and Jakob Nikolas Kather. Artificial intelligence agents in cancer research and oncology. *Nature Reviews Cancer*, 26:256–269, 2026. doi: 10.1038/s41568-025-00900-0.

Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal LLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13084–13094, 2024. doi: 10.1109/CVPR52733.2024.01243.

Feng Xu, Chuang Zhu, Wenqi Tang, Ying Wang, Yu Zhang, Jie Li, Hongchuan Jiang, Zhongyue Shi, Jun Liu, and Mulan Jin. Predicting axillary lymph node metastasis in early breast cancer using deep learning on primary tumor biopsy slides. *Frontiers in Oncology*, 11:759007, 2021. doi: 10.3389/fonc.2021.759007.

Chunze Yang, Qidong Liu, Wenjie Zhao, Yue Tang, Jiusong Ge, Di Zhang, Jiashuai Liu, Lei Wu, Junbo Lu, Ni Zhang, Xian Wu, Zeyu Gao, and Chen Li. PathNavigate: A training-free pathology agent with surprise-guided scan and shared slide memory for whole-slide image VQA. *arXiv preprint arXiv:2605.23559*, 2026. URL https://arxiv.org/abs/2605.23559.

Chengyang Zhang, Wenchuan Zhang, Bo Li, Xinyu Liu, Jiaming Yang, Mengran Li, Chenxun Deng, Jie Chen, Yang Zhang, Wei Ju, Yuhao Yi, Hong Bu, and Jiancheng Lv. Do pathology vision-language models truly see pathology? *arXiv preprint arXiv:2607.21065*, 2026a. URL https://arxiv.org/abs/2607.21065.

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, 2026b. URL https://arxiv.org/abs/2505.11404.

A Detailed Rescue–Route–Rescore Method

A.1 Detailed Overview

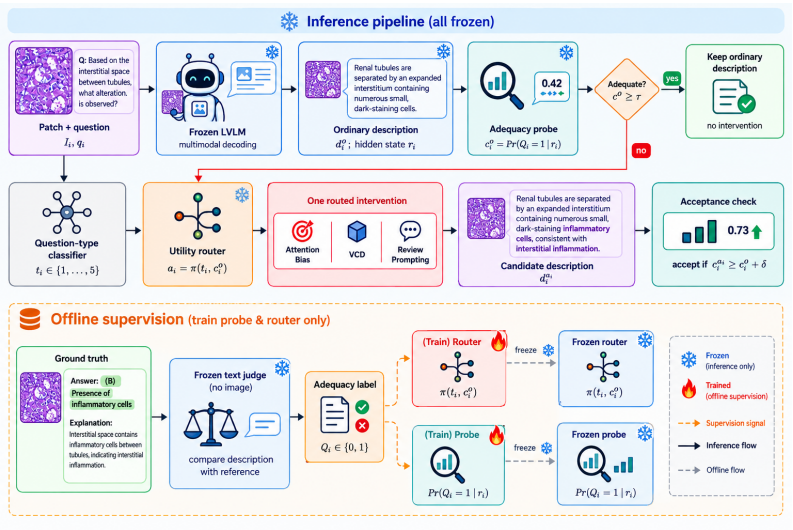


Figure 5: **From patch-level supervision to frozen test-time correction.** A reference-conditioned text judge supplies offline adequacy labels for fitting the probe and router (bottom). At inference (top), the probe gates the ordinary description, the router chooses one intervention, and clean rescoring checks whether to keep its candidate. The schematic's $a_i = \pi(t_i, c_i^0)$ abbreviates the action a_i^* selected from Equation 4, not the full gated policy. No judge or reference is available online.

Following Section 3, WSI S_n has pyramid $\mathcal{P}_n = \{S_n^{(\ell)}\}_{\ell=1}^{L}$, with patches $\{I_{n,\ell,j}\}_{j=1}^{J_{n,\ell}}$ at level ℓ. Let t index an individual perceptor call for slide n. The host agent selects a patch identified by $(\ell_{n,t}, j_{n,t})$. The local notation then corresponds to

$$i = (n, t), \qquad I_i = I_{n,\ell_{n,t},j_{n,t}}, \qquad q_i = q_n, \tag{6}$$

so that

$$d_i^0 = d_{n,t}^0 = f_m(I_i, q_i). \tag{7}$$

Several perceptor calls may belong to the same acquisition batch or reasoning round. The controller returns the selected description $\hat{d}_i$ for downstream use, while patch acquisition follows the host agent's policy.

Appendix Fig. 5 summarizes the three decisions: gate, route, and accept. Using the feature vector and utility in Equations 3– 4, the pre-acceptance policy is

$$\pi_m(i) = \begin{cases} \text{retain}, & c_i^0 \geq \tau, \\ a_i^* = \arg\max_{a \in \mathcal{A}} u_{i,a}, & c_i^0 < \tau \end{cases} \tag{8}$$

Here $c_i^0 \in [0, 1]$ is the calibrated adequacy confidence of the ordinary description and τ is the global gate. Retention is a gate or acceptance outcome, not an additional fitted router arm.

After a_i^* generates candidate $d_i^{a_i^*}$, clean rescoring produces confidence $c_i^{a_i^*}$ and the final description is

$$\hat{d}_i = \begin{cases} d_i^{a_i^*}, & c_i^{a_i^*} \geq c_i^0 + \delta, \\ d_i^0, & \text{otherwise}, \end{cases} \tag{9}$$

where δ is a single shared acceptance margin. The implementation compares $\delta \in \{-\infty, 0, 0.02, 0.04, 0.06, 0.08, 0.1, 0.12\}$, corresponding to accept-all, non-decreasing confidence, and a δ -point margin in a validation set (for inference we chose $\delta = 0.08$). Only the router is learned to choose actions; the acceptor is a fixed comparison.

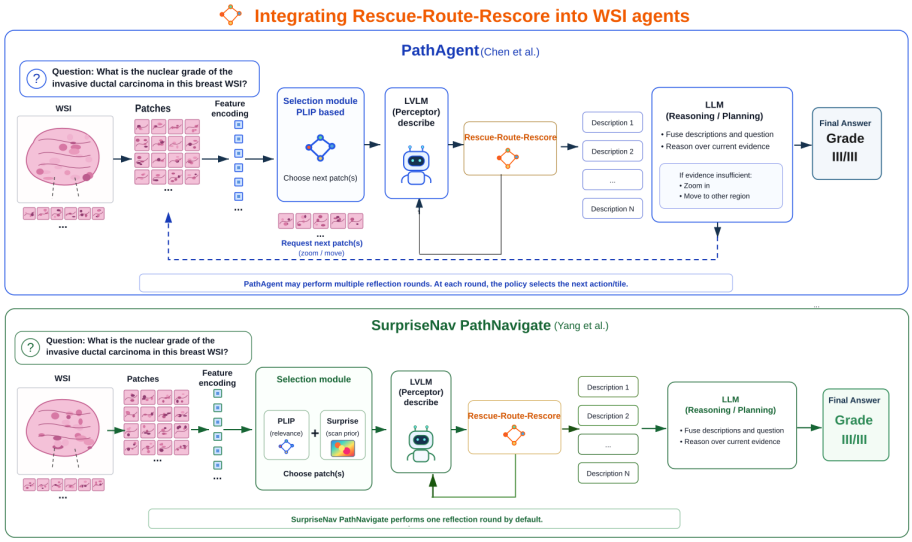


Figure 6: **The frozen controller is inserted at each WSI perceptor call.** In PATHAGENT (top), an accepted description can influence later retrieval and magnification. In the evaluated one-reflection SURPRISENAV configuration (bottom), it mainly changes the evidence available to the final reasoner. PLIP(Huang et al., 2023)is a CLIP(Radford et al., 2021) based vision-language encoder model for pathology - it helps to narrow down the location of a specific language based query in the WSI. The controller is transferred unchanged and receives no PathMMU reference, WSI label, or trajectory feedback.

A.2 MODEL-SPECIFIC CALIBRATED CONFIDENCE

Here r_i, Q_i, and c_i denote the representation, adequacy label, and estimated adequacy probability of a generic description. Superscripts 0 and a distinguish ordinary and intervention-generated descriptions, respectively.

Representation collection. Each LVLM has a separate probe because its hidden space and generation format differ. We greedily generate at most 512 new tokens, strip terminal padding and EOS tokens, and run the complete prompt-response sequence through an ordinary forward pass with interventions disabled. The feature $r_i = h_{p_i}^{(L)}$ is taken from the last decoder layer at the last token inside `<answer>` tags; when answer tags are absent or malformed, p_i falls back to the last non-special response token.

Prompt bank and adequacy target. While generating the judge LLM derived adequacy targets each base image-question item in the PathMMU development set is rendered twice: once with the canonical detailed SURPRISENAV or PATHAGENT prompts and once with one of four frozen paraphrased, compressed, mixed-style, and reordered variants derived from those two. The alternative is fixed for each base item. All variants preserve the image, question, options, and requested description task, while varying instruction length, ordering, emphasis, and style. They request visible evidence and uncertainty and prohibit selecting an answer option. Neither the correct answer nor the reference explanation is exposed to the LVLM.

This controlled prompt variation is meant to serve as nuisance normalization for the adequacy probe. As mentioned in the main text REVPROMPT later appends a neutral review procedure and thus changes the prompt context under which a candidate description is generated and rescored. Although the exact evolved procedures are not present during probe fitting, the prompt bank exposes the probe to related changes in verbosity, ordering, and descriptive scaffolding. The probe is therefore encouraged to predict adequacy from response content that is stable across prompt formulations,

rather than from one particular instruction template. All prompt variants belonging to the same image remain in the same grouped split, so this prompt robustness is learned without leaking an image across training, calibration, or evaluation folds. We consequently describe the probe as only approximately robust to prompt conditioning, rather than strictly prompt-invariant.

After this, a frozen MedGemma-27B text judge receives the question, options, correct answer, reference explanation, and candidate description, but not the source image. It scores relevance, reference consistency, coverage, unsupported claims, and overall quality on 0–4 scales, together with a binary instruction-following flag. The deterministic target is

$$\begin{aligned} Q_i = 1 \quad \Longleftrightarrow \quad & \mathrm{rel}_i \geq 3, \quad \mathrm{cons}_i \geq 3, \quad \mathrm{cov}_i \geq 2, \\ & \mathrm{unsupported}_i \leq 1, \quad \mathrm{follow}_i = 1, \quad \mathrm{quality}_i \geq 3. \end{aligned} \tag{10}$$

Thus the target is reference-conditioned description adequacy, not image-aware factual verification or final-answer correctness. The probe is trained only on Q_i.

Probe and calibration. For LVLM m, let $r_i \in \mathbb{R}^{d_m}$ be the final-layer hidden state extracted for description i as specified above. We first standardize every hidden dimension using the probe-training mean $\mu_m \in \mathbb{R}^{d_m}$ and standard deviation $\rho_m \in \mathbb{R}^{d_m}$:

$$\widetilde{r}_i = (r_i - \mu_m) \oslash \rho_m, \tag{11}$$

where $\oslash$ denotes element-wise division. A compact one-hidden-layer MLP then maps this standardized representation to an uncalibrated adequacy logit:

$$\ell_i = g_m(\widetilde{r}_i) = w_{2,m}^{\top} \mathrm{GELU}(W_{1,m}\widetilde{r}_i + b_{1,m}) + b_{2,m}. \tag{12}$$

Here $W_{1,m} \in \mathbb{R}^{32 \times d_m}$ and $b_{1,m} \in \mathbb{R}^{32}$ define the 32-dimensional hidden layer, $w_{2,m} \in \mathbb{R}^{32}$ is the output vector, and $b_{2,m}$ is the output bias.

The scalar logit ℓ_i is converted into calibrated adequacy confidence by an orientation-preserving Platt map:

$$c_i = \mathrm{sigmoid}(\lambda_m \ell_i + \kappa_m) \approx \Pr(Q_i = 1 \mid r_i), \tag{13}$$

where $\lambda_m = \mathrm{softplus}(\widetilde{\lambda}_m) > 0$ is the fitted Platt slope and κ_m is its intercept. Constraining λ_m to be positive preserves the probe's ordering: a larger uncalibrated adequacy logit cannot be mapped to a lower calibrated confidence.

The MLP parameters are optimized with full-batch binary cross-entropy and AdamW (learning rate 10^{-3}, weight decay 10^{-3}, at most 400 epochs, patience 40, seed 42). Within each outer training partition, four of five grouped folds are allocated to probe fitting, and the remaining fold is reserved for Platt calibration. An additional grouped validation split within the probe-fitting portion selects the epoch count. We then refit the MLP on the entire probe-fitting portion, including this inner validation set, for the selected number of epochs. Finally, we freeze the MLP and fit the Platt parameters (λ_m, κ_m) on the separate calibration fold, which is excluded from both epoch selection and MLP training.

Fitting and testing. Groups are defined by PathMMU domain and image name, keeping every question and prompt variant from an image together. Development performance uses five outer grouped folds. Within each outer training partition, one of five grouped image folds is reserved exclusively for Platt calibration. The remaining groups are used for probe fitting. The outer test groups are used only to obtain predictions.

The saved MLP, feature statistics, and Platt parameters are evaluated once on the descriptions in the held out testing split. We report AUROC, average precision, Brier score, log loss, ten-bin ECE and reliability. Router training uses only development out-of-fold (OOF) confidence; validation, external testing, and WSI deployment use the frozen checkpoint.

The gate uses c_i^0 to focus compute on descriptions likely to be inadequate; the acceptor compares c_i^a with c_i^0. Crucially, intervention mechanisms are disabled during both scoring passes (except for review prompting). This common, unmodified measurement path avoids scoring an attention-biased or contrastive-decoding state with a probe fitted on ordinary states.

A.3 QUESTION-ONLY TASK TAXONOMY

Every question is assigned to exactly one of five task types:

1. morphology/localization;
2. diagnosis/classification;
3. grade/stage;
4. biomarker/immunohistochemistry (IHC); and
5. clinical/knowledge.

A frozen Qwen2.5-7B-Instruct question-only classifier produces the assignment. It never receives the image, base prediction, candidate response, or ground truth. This restriction prevents the task classifier from leaking outcome information into the router. The same taxonomy and classification prompt are used for all four LVLMs and in both WSI agents.

A.4 CANDIDATE INTERVENTIONS

Since the study was focused on evaluating the usefulness of the local evidence quality for controlling the LVLMs, the intervention set is intentionally small and fixed before final evaluation. We do not search over large intervention-specific hyperparameter grids.

Attention biasing. Following Liu et al. (2024), our ONLY-ATTNB arm within the agentic pipelines isolates an attention-only perceptual intervention. At decoder layer ℓ, attention head h, generation position t, and image-token position j, let

$$z_{\ell htj} = \frac{q_{\ell ht}^{\top} k_{\ell hj}}{\sqrt{d_h}} + M_{tj} \tag{14}$$

denote the scalar attention score before the softmax. Here $q_{\ell ht}$ is the query for the token currently being generated, $k_{\ell hj}$ is the key of image token j, d_h is the head dimension, and M_{tj} is the usual causal or padding mask. For the final query row and every image-token key, ONLY-ATTNB applies

$$z'_{\ell htj} = z_{\ell htj} + \alpha_{\text{attn}} |z_{\ell htj}|, \qquad \alpha_{\text{attn}} = 0.2. \tag{15}$$

The update is applied to all decoder attention heads from layer 2 through the final decoder layer; the vision tower is unchanged. Hooks are active only during candidate generation. They are disabled during ordinary rescoring, so the probe scores the generated description using an intervention-free forward pass.

Visual contrastive decoding. Following Sicong Leng (2023), VCD contrasts next-token predictions from the original image with those from a Gaussian-blurred view of the same image (blur radius 10). At decoding step t, let $z_{t,k}^{\text{orig}}$ be the frozen LVLM's output logit for vocabulary token k given the original image, and let $z_{t,k}^{\text{blur}}$ be the corresponding logit from the blurred-image branch. Their contrastive logit is

$$z_{t,k}^{\text{VCD}} = (1 + \alpha_{\text{VCD}}) z_{t,k}^{\text{orig}} - \alpha_{\text{VCD}} z_{t,k}^{\text{blur}}, \qquad \alpha_{\text{VCD}} = 1.0. \tag{16}$$

This increases the relative score of tokens supported more strongly by the original image than by its blurred counterpart and works to reduce the dependence on language priors for question answering.

To prevent the subtraction from promoting an implausible token, token k remains eligible only when

$$z_{t,k}^{\text{orig}} \geq \max_{v} z_{t,v}^{\text{orig}} + \log \beta, \qquad \beta = 0.1. \tag{17}$$

Thus only tokens within approximately 2.30 logit units of the original branch's most likely token are considered; all others are masked before token selection. The resulting description is subsequently rescored through an ordinary forward pass using the original image and no contrastive processor.

Review Prompting REVPROMPT uses frozen Qwen2.5-7B-Instruct as a text-only meta-optimizer. For an eligible example it sees the question, answer options, magnification, ordinary description, and c_i^0, but neither the image nor ground truth. It generates at most 800 tokens and proposes a population of two neutral review procedures, each 20–140 words with two to four symmetric checks (e.g., whether a feature is present or absent). The procedure strategies are architecture, cytology, staining, image quality, and counterevidence audits. The perceptor re-reads the same image once per procedure. The highest-confidence candidate is accepted only if $c_i^a \geq c_i^0 + 0.08$.

A.5 INTERVENTION GATE

For the routing study we evaluate only $\tau \in \{0.4, 0.5, 0.6, 0.8\}$ and select one global threshold ($\tau = 0.6$) using pooled validation adequacy across all four models. The default intervention threshold in the fixed runs is $\tau = 0.6$.

A.6 INTERPRETABLE TASK-CONDITIONED ROUTER

Equations 3–4 are the router definition. The logistic regressions are fitted by 60 steps of iteratively reweighted least squares with L_2 penalty 1.0 on all non-intercept coefficients.

To interpret a fitted model, partition its coefficient vector as

$$w_{m,a} = \left[w_{m,a,0},\ w_{m,a,c},\ w_{m,a,\text{task}}^\top,\ w_{m,a,\text{int}}^\top\right]^\top. \tag{18}$$

For a fixed question type t, only the following two sums affect predictions:

$$B_{m,a,t} = w_{m,a,0} + w_{m,a,\text{task},t}, \qquad S_{m,a,t} = w_{m,a,c} + w_{m,a,\text{int},t}. \tag{19}$$

Thus the candidate-adequacy logit is $w_{m,a}^\top x_i = B_{m,a,t_i} + S_{m,a,t_i} c_i^0$. We interpret these effective task intercepts B and confidence slopes S, rather than individual raw coefficients. This also resolves the redundancy between the global intercept and the exhaustive task indicators.

Exact utility ties are broken in the predeclared low-to-high-cost order ATTNB, then VCD, then REVPROMPT. This order reflects the required work— an attention-modified generation, an additional blurred-image branch, and a meta-model call plus multiple candidate generations, respectively—and is fixed independently of evaluation accuracy.

A.7 RESCORING BASED ACCEPTANCE

For a candidate description, *rescoring* is a second, full teacher-forced forward pass over the prompt and generated response. For ATTNB, the attention hook is disabled; for VCD, rescoring uses the original image without the blurred branch or contrastive processor. REVPROMPT also uses ordinary attention and decoding, but retains the revised instruction that formed the candidate's generation context. Its rescored confidence score is therefore mechanism-free but prompt-conditioned. In every case, the acceptor receives no intervention-specific feature. We compare the following prospectively specified rules on pooled validation data:

1. accept every valid intervention;
2. accept if $c_i^a \geq c_i^0$; and
3. accept if $c_i^a \geq c_i^0 + \delta$ with $\delta = \{0.02, 0.04, 0.06, 0.08, 0.1, 0.12\}$).

The selected rule ($\delta = 0.08$) is frozen and shared across interventions, models, and WSI pipelines.

B DETAILED EXPERIMENTAL DESIGN

This appendix records the full evaluation grid and frozen configurations.

B.1 DATA PREPARATION AND SPLIT ASSIGNMENT

TCGA conversion to OME-Zarr. After acquisition, each TCGA whole-slide image (WSI) was indexed in a flat manifest containing its unique slide identifier, source path, and destination path. We

converted the source slides into independent `.ome.zarr.zip` archives with OME-NGFF v0.5 multiscale metadata in a Zarr v3 store. The fixed-resolution pyramid starts at $0.5\ \mu\mathrm{m/pixel}$ and contains up to five levels. RGB images are stored as unsigned 8-bit arrays using 512×512-pixel chunks, 1024×1024-pixel shards, and JPEG compression at quality 80 with 4:2:2 chroma subsampling.

Conversion is parallelized across slides. The complete input manifest is retained as `slides.parquet`; `metadata.yaml` records the conversion settings and source-code revision; and `slides_metadata.parquet` and `conversion_results.csv` record source dimensions, resolution levels, scanner metadata, output paths, read failures, and per-slide status. Destination paths are supplied by the manifest; for SlideBench-TCGA, slides are organized as `wsi/<project>/<barcode>.<uuid>.ome.zarr.zip`. During benchmark preparation, each bare TCGA barcode in the question file is resolved to this full slide identifier.

PathMMU development, validation, and test sets. PathMMU(Sun et al., 2024) contains the PubMed, SocialPath, EduContent, Atlas, and PathCLS domains. It is important to distinguish the released partition names from their roles in our study (the constructed *development set*, the released `val` partition as the *validation set*, and `test_tiny` as the *final test set*). The official test benchmark contains 9,660 questions in a split called `test_all` and describes `test_tiny` as a subset extracted from that set; the released data expose 1,139 under `test_tiny`. To prevent this provenance from introducing leakage, we did not use the released `test_all` key unchanged. Before any probe or router fitting, we removed every `test_all` record whose image group, defined by domain and image name, also appeared in `test_tiny` resulting in 8521 records in the "development set". Accordingly, "development set" below means the remainder of `test_all` after this image-group exclusion, while all 1,139 `test_tiny` questions are reserved for final frozen evaluation. Questions and prompt variants derived from the same retained image remain grouped throughout fitting. The 710-question validation set, associated with 510 images, is used to select the shared confidence gate and acceptance rule. The official PathMMU training partition is not used in these experiments. Table 5 summarizes this role assignment.

Table 3: Factorial evaluation grid. The learned router is reported separately from the four fixed inference conditions.

Factor	Levels
Dataset	SlideBench-VQA-TCGA; SlideBench-VQA-BCNB; WSI-VQA
Pipeline	PATHAGENT; SURPRISENAV
Perceptor LVLM	Quilt-LLaVA 7B; Patho-R1 3B; Patho-R1 7B; MedGemma 4B
Fixed condition	Baseline; ONLY-ATTNB; ONLY-VCD; ONLY-REVPROMPT

Table 4: Pathology LVLMs. Parameters remain frozen for every intervention.

Model	Size	Backend	Output convention
Patho-R1	3B	Qwen2.5-VL	Native `<think>/<answer>` reasoning
Patho-R1	7B	Qwen2.5-VL	Native `<think>/<answer>` reasoning
Quilt-LLaVA v1.5	7B	LLaVA-1.5	Direct description; 576 image tokens
MedGemma 1.5	4B	Gemma 3	Direct description

Table 5: PathMMU protocol. Named splits are the keys in the released `data.json`

Study role	Source named split	Questions	Purpose
Development	`test_all`	9660 (pre-excl.)	Exclude all `test_tiny` image groups, then perform nested image-grouped OOF probe scoring and router fitting
Validation	`val`	710	Select the shared confidence gate and acceptance rule
Final test	`test_tiny`	1,139	Frozen probe and paired policy evaluation

Table 6: Patch-level policy arms. Eligible examples fall below the frozen confidence gate.

Policy	Definition
Normal	No intervention
Always ATTNB	Apply ATTNB to every eligible response
Always VCD	Apply VCD to every eligible response
Always REVPROMPT	Apply review prompting to every eligible response
Best fixed action	Use one validation-selected action for all eligible responses
Task lookup	Select one validation-best action separately for each question type; ignore within-type confidence
Confidence-only	Fit the same per-action logistic utilities using confidence alone
Router	Add task indicators and confidence–task interactions to the per-action utilities
Router + acceptance	Route once, then retain the candidate only if rescored confidence meets the frozen margin

Table 7: Frozen agent configurations used for all perceptor comparisons.

Component	PATHAGENT	SURPRISENAV
Selection	Top-200 PLIP pool; 10% initial, 5% replenish	PLIP query navigation; $\alpha_q = 0.5$; top-10 jumps
Evidence	Top-5 descriptions per round	Top-2 patches per ROI
Control	Up to 5 attempts; sufficient/zoom/retrieve	One reflection round
Budget	At most 7 perceptor calls	At most 7 perceptor calls
Scale	5× base; top-2 regions at 20×	CONCH 5×/20×; local surprise
Patch	512 pixels	512-pixel feature-aligned tiles
Adjudicator	Qwen3.5-4B	Qwen3.5-4B

PathAgent embeds the executor's missing-information request with PLIP to select additional patches and can zoom into the top two regions. SurpriseNav uses a 768-dimensional query-guided surprise representation, up to 7 jumps, two patches per region of interest(ROI), and one reflection round. TCGA(The Cancer Genome Atlas Research Network et al., 2013) and WSI-VQA are read from OME-Zarr; BCNB uses flat JPEG fields at native 20×. Precomputed CONCH(Lu et al., 2024) features supply 5× and 20× views. Retrieval-augmented generation and MCQ debiasing are disabled unless explicitly stated.

C FULL FIXED-INTERVENTION RESULT MATRICES

Table 8: Oriented AUROC for frozen-judge adequacy on PathMMU on a signal screening subset ($n = 1000$ per LVLM).

Signal	Patho-R1 3B	Patho-R1 7B	Quilt-LLaVA 7B	MedGemma 4B
Adequacy confidence	**0.739**	**0.682**	**0.684**	**0.800**
Blur confidence drop	0.568	0.554	0.552	0.546
SVAR(Jiang et al. (2025))	0.591	0.579	0.539	0.510
PAS(Hoang et al., 2026)	0.556	0.571	0.504	0.510
HIDE(Chatterjee et al., 2025)	0.521	0.499	0.458	0.500

For the blur-confidence-drop diagnostic, we generated one description from the original image and a second description using the same question, prompt, LVLM, and greedy decoding after applying an obliterative blur. The perturbation downsamples the image to 16×16 pixels, restores it to its original size, and applies a Gaussian blur with radius 32 pixels, removing cellular and tissue morphology while retaining coarse color and spatial layout. The same frozen adequacy probe scores both generated descriptions. For item i, the signal is

$$B_i = c_i^{\text{original}} - c_i^{\text{blurred}}. \tag{20}$$

A positive value means that the probe assigns less adequacy confidence when the LVLM can no longer inspect morphology; a value near zero indicates that the confidence is insensitive to this

removal. Table 8 reports the oriented AUROC of B_i for discriminating judge-adequate from judge-inadequate descriptions.

Table 9: Overall-accuracy matrix for SlideBench-VQA-TCGA (n = 2,629).

Pipeline	Perceptor LVLM	Baseline	ONLY-ATTNB	ONLY-VCD	ONLY-REVPROMPT	Router+Accept
PATHAGENT	Quilt-LLaVA 7B	38.84	39.63	41.84	39.03	42.11
PATHAGENT	Patho-R1 7B	39.10	50.21	48.88	49.87	52.34
PATHAGENT	Patho-R1 3B	40.32	45.04	44.24	44.43	47.70
PATHAGENT	MedGemma 4B	35.49	38.04	38.65	37.81	40.28
SURPRISENAV	Quilt-LLaVA 7B	36.82	38.15	37.47	36.21	38.57
SURPRISENAV	Patho-R1 7B	42.15	47.09	45.64	46.33	48.73
SURPRISENAV	Patho-R1 3B	39.67	41.0	42.37	40.51	42.37
SURPRISENAV	MedGemma 4B	35.41	34.69	37.16	36.48	39.02

Table 10: Overall-accuracy matrix for SlideBench-VQA-BCNB (n = 7,267).

Pipeline	Perceptor LVLM	Baseline	ONLY-ATTNB	ONLY-VCD	ONLY-REVPROMPT	Router+Accept
PATHAGENT	Quilt-LLaVA 7B	39.81	40.11	42.07	40.54	43.42
PATHAGENT	Patho-R1 7B	43.76	47.20	46.03	46.68	48.59
PATHAGENT	Patho-R1 3B	44.90	45.91	46.17	40.61	47.35
PATHAGENT	MedGemma 4B	38.59	40.18	39.36	39.03	41.27
SURPRISENAV	Quilt-LLaVA 7B	38.68	40.42	39.07	40.02	41.48
SURPRISENAV	Patho-R1 7B	41.74	45.18	43.92	44.61	45.34
SURPRISENAV	Patho-R1 3B	42.80	44.05	43.62	43.31	44.28
SURPRISENAV	MedGemma 4B	36.95	36.19	38.43	37.84	39.60

Table 11: overall-accuracy matrix for WSI-VQA (n = 701 per row).

Pipeline	Perceptor LVLM	Baseline	ONLY-ATTNB	ONLY-VCD	ONLY-REVPROMPT	Router+Accept
PATHAGENT	Quilt-LLaVA 7B	28.39	27.82	30.10	29.24	31.24
PATHAGENT	Patho-R1 7B	31.24	33.10	32.10	32.67	34.24
PATHAGENT	Patho-R1 3B	29.67	28.96	30.39	30.81	31.95
PATHAGENT	MedGemma 4B	30.67	28.82	29.39	29.67	30.81
SURPRISENAV	Quilt-LLaVA 7B	23.82	24.39	25.82	24.96	26.82
SURPRISENAV	Patho-R1 7B	26.68	28.10	27.25	27.67	29.10
SURPRISENAV	Patho-R1 3B	28.39	25.25	26.53	26.82	28.96
SURPRISENAV	MedGemma 4B	24.82	23.40	26.11	26.11	27.25

Table 12: **Adequacy versus answer correctness as a controller-training target.** The correctness- and adequacy-supervised controllers use the same architecture, features, candidate cache, data splits, and validation procedure, differing only in their training labels. All controllers are frozen before transfer to the WSI agents. Entries report final-answer accuracy (%), macro-averaged over the four LVLMs.

	PATHAGENT			SURPRISENAV		
Controller supervision	SlideBench-TCGA	SlideBench-BCNB	WSI-VQA	SlideBench-TCGA	SlideBench-BCNB	WSI-VQA
Ordinary inference	38.44	41.77	29.99	38.51	40.04	25.93
Answer correctness	41.27	43.12	30.45	40.58	40.80	27.32
Description adequacy	**45.61**	**45.16**	**32.06**	**42.16**	**42.68**	**28.03**

D FULL CONFIDENCE-PROBE RESULTS

Table 13: Observed confidence-probe performance for reference-conditioned, frozen-judge description adequacy. Dev OOF denotes nested image-grouped out-of-fold predictions on a PathMMU development split; the Platt calibrator for each outer fold uses separate image groups. External applies the final frozen MLP and calibrator to a disjoint PathMMU test split.

Model	Evaluation	Adeq. (%)	AUROC	Brier	ECE
Patho-R1 3B	Dev OOF	58.11	0.682	0.219	0.006
	External	62.2	0.705	0.205	0.034
Patho-R1 7B	Dev OOF	60.73	0.663	0.219	0.014
	External	65.6	0.687	0.205	0.038
Quilt-LLaVA 7B	Dev OOF	50.38	0.664	0.228	0.015
	External	47.6	0.678	0.225	0.025
MedGemma 4B	Dev OOF	43.42	0.797	0.180	0.019
	External	47.5	0.779	0.186	0.044

The development and external test rows contain 8521 and 1139 base PathMMU items, respectively. All questions and prompt sharing an image remain in the same group, and images shared with the external test split are excluded before development fitting.

External AUROC remains within 0.024 of OOF AUROC for every model. ECE rises from 0.006–0.019 OOF to 0.025–0.044 externally (see Fig. 7).

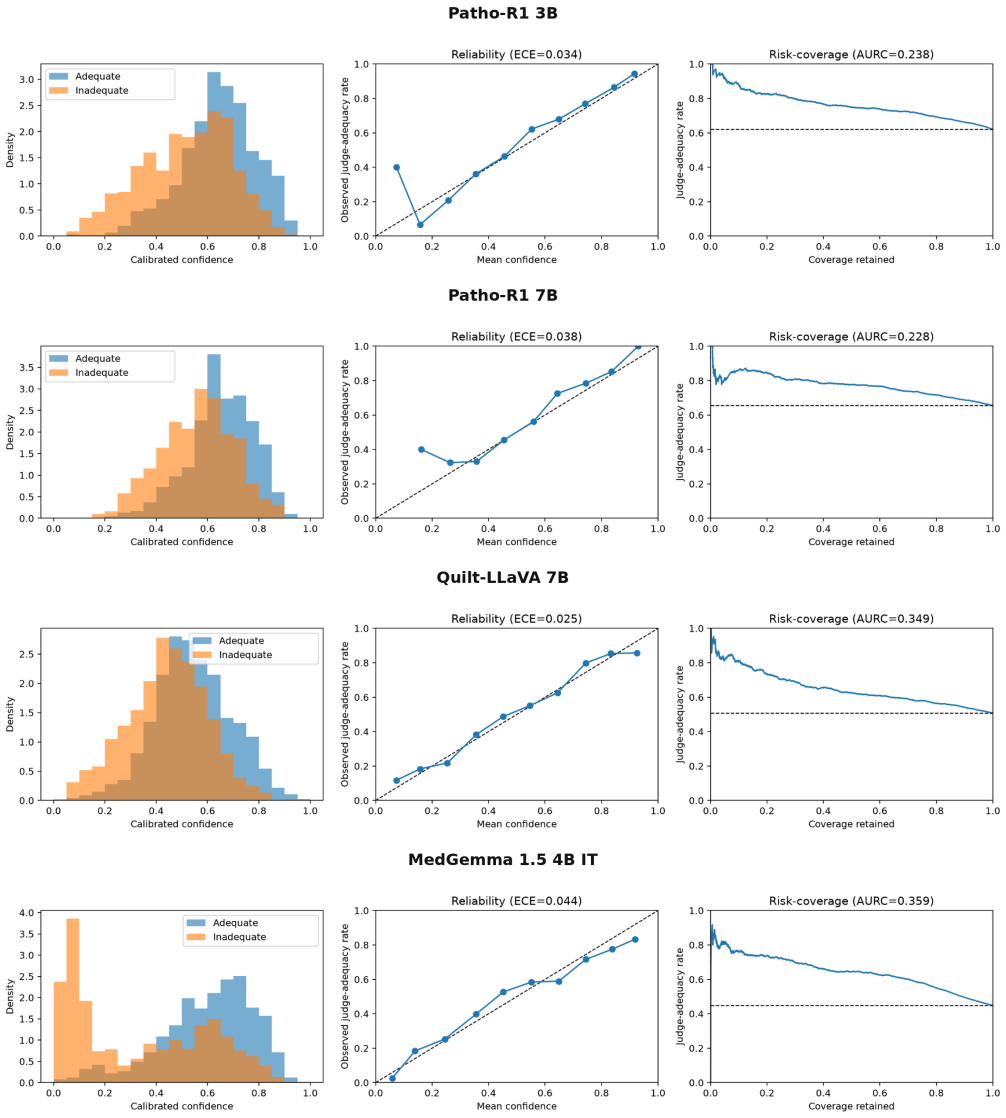


Figure 7: Absolute Confidence distributions for the four LVLMS with reliability and risk-coverage on external.

E Expanded Router and Transfer Results

This section examines why confidence- and task-conditioned routing improves over applying one intervention globally. We connect four complementary pieces of evidence. First, Table 14 compares the learned router against fixed and partially conditioned policies, reporting both description adequacy and downstream LLM reader answer accuracy. Second, Table 15 tests whether a router fitted for one LVLM transfers to another. Third, Fig. 9 shows that intervention effects vary across models and question types. Finally, Fig.s 10 and 4 show how those effects vary with the adequacy confidence of the ordinary description and how the fitted router converts these patterns into decision regions. Together, these analyses explain both the value of routing and the benefit of fitting a separate router for each LVLM. All results are computed from the same immutable candidate cache used in the main experiments.

The adequacy intervals in Table 14 were computed on the frozen evaluation split using a nonparametric grouped bootstrap with 2,000 replicates and random seed 42. The resampling unit was the

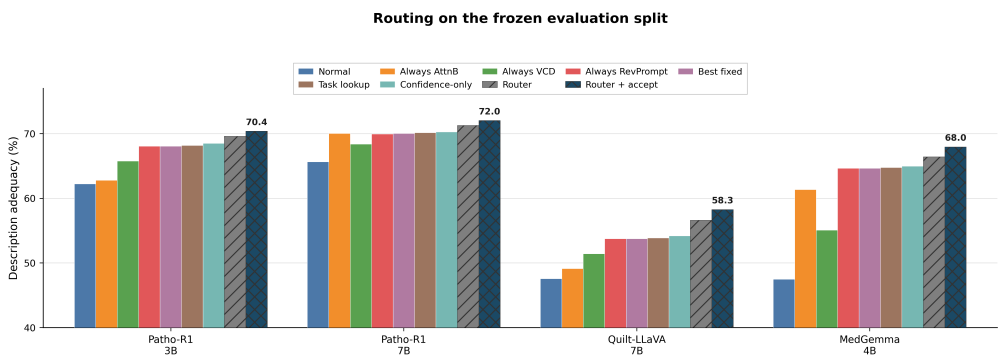


Figure 8: Description adequacy on the frozen PathMMU evaluation split. *Best fixed* uses one validation-selected intervention for all gated cases, while *task lookup* chooses separately for each question type. *Confidence-only* omits task features. *Router+accept* has the highest measured adequacy for all four LVLMs.

Table 14: Expanded policy comparison on PathMMU (%). Parentheses contain 95% confidence intervals for description adequacy; the supplied summary does not contain confidence intervals for downstream agent accuracy or macro-averages. Harm is reported on the adequacy row and denotes the conditional harm rate for +Accept, $100 \times \text{harmed}/\text{changed}$.

Model / outcome	Normal	Best fixed	Conf.-only	Task lookup	Router	+Accept	+Accept harm
Patho-R1 3B / adequacy	62.2 (59.2–65.4)	68.1 (65.0–71.0)	68.5 (65.4–71.4)	68.1 (65.1–71.0)	69.6 (66.5–72.5)	70.4 (67.4–73.3)	18.0
Patho-R1 3B / answer acc.	65.1	66.0	66.5	66.0	68.0	69.1	–
Patho-R1 7B / adequacy	65.6 (62.4–68.7)	69.9 (66.9–72.8)	70.2 (67.2–73.0)	70.4 (67.4–73.2)	71.2 (68.2–74.0)	72.0 (69.0–74.8)	20.0
Patho-R1 7B / answer acc.	65.5	66.3	66.6	67.0	68.4	69.3	–
Quilt-LLaVA 7B / adequacy	47.6 (44.3–50.8)	51.4 (48.2–54.8)	54.2 (50.8–57.6)	54.4 (51.0–57.8)	56.6 (53.3–59.8)	58.2 (54.9–61.4)	24.0
Quilt-LLaVA 7B / answer acc.	43.3	47.4	47.5	48.3	50.1	51.2	–
MedGemma 4B / adequacy	47.5 (44.3–50.7)	61.3 (58.1–64.6)	64.6 (61.5–67.9)	62.2 (59.0–65.4)	65.8 (62.7–68.8)	68.0 (64.9–70.9)	8.0
MedGemma 4B / answer acc.	53.7	56.3	56.6	57.2	58.4	59.6	–
Macro-average / adequacy	55.7	62.7	64.4	63.8	65.8	67.2	17.5
Macro-average / answer acc.	56.9	59.0	59.3	59.6	61.2	62.3	–

unique PathMMU image (identified by domain and image name), sampled with replacement; all questions associated with a sampled image remained together. Adequacy was recomputed for each policy in every replicate, and the reported limits are the 2.5th and 97.5th percentiles of the resulting distribution. The intervals therefore only quantify finite-test-sample uncertainty while accounting for within-image dependence among questions.

E.1 Policy outcomes and the adequacy–accuracy relationship

Table 14 shows a consistent progression from unconditional to increasingly instance-specific policies. Averaged equally over the four LVLMs, the ordinary policy obtains 55.7% description adequacy and 56.9% answer accuracy. Applying the best single intervention for each model raises these values to 62.7% and 59.0%, respectively. Conditioning on confidence or task type provides further, but incomplete, gains. The full router reaches 65.8% adequacy and 61.2% answer accuracy, while rescoring acceptance raises them to 67.2% and 62.3%. Thus, relative to the best fixed intervention, router+accept improves macro-averaged adequacy by 4.5 pp and answer accuracy by 3.3 pp.

The advantage of the full router over either partial policy supports the use of both task type and baseline confidence. Relative to the task-only lookup, the router improves adequacy by 0.8, 1.5, 2.2, and 3.6 pp for Patho-R1 7B, Patho-R1 3B, Quilt-LLaVA 7B, and MedGemma 4B, respectively. The corresponding answer-accuracy gains are 1.4, 2.0, 1.8, and 1.2 pp. Task identity alone therefore does not determine the best treatment: the appropriate action can change as the confidence of the ordinary description changes. Conversely, the differences between confidence-only and task-lookup policies show that confidence alone also cannot capture the task-specific treatment effects.

Rescoring based acceptance further improves every model rather than merely improving the macro-average. Compared with the pre-acceptance router, it adds between 0.8 and 2.2 adequacy pp and

between 0.9 and 1.2 answer-accuracy pp. This is consistent with its intended role: routing chooses a promising intervention, whereas acceptance rejects candidates that fail to improve upon the ordinary description when both are rescored under standard inference. The reported harm column nevertheless shows that acceptance is not perfect, with residual harm ranging from 8.0% to 24.0% among accepted replacements. Accordingly, the acceptance mechanism should be interpreted as a risk-reduction filter rather than a guarantee of improvement.

The table also provides an intervention-based counterpart to the association studied in RQ1. From the ordinary policy to router+accept, adequacy and answer accuracy increase in the same direction for every LVLM:

LVLM	Δ adequacy (pp)	Δ answer accuracy (pp)
Patho-R1 3B	+8.2	+4.0
Patho-R1 7B	+6.4	+3.8
Quilt-LLaVA 7B	+10.6	+7.9
MedGemma 4B	+20.5	+5.9

This aligned movement supports the central premise that a more adequate description is generally more useful to the answerer. The relationship is not one-to-one, however. MedGemma, for example, gains 20.5 adequacy pp but only 5.9 answer-accuracy pp. This attenuation is expected: an adequate description does not guarantee that the downstream answerer will reason correctly, while an inadequate description can occasionally still permit a correct answer through partial evidence or guessing. Adequacy is therefore a useful transferable intermediate target, but not a replacement for reporting answer accuracy. Thus, Table 14 provides complementary evidence that policies optimized for local adequacy produce downstream answer gains across all four LVLMs.

Table 15: Router+accept cross-model transfer matrix. Each cell reports the target-model description adequacy relative to that target's best fixed action, in percentage points. Rows identify the model used to train the router and columns identify the target model.

Source router+accept \ target	Patho 3B	Patho 7B	Quilt-LLaVA	MedGemma
Patho-R1 3B	+2.30	+1.70	+3.80	+4.80
Patho-R1 7B	+1.80	+2.10	+3.40	+5.20
Quilt-LLaVA 7B	−0.60	−1.3	+6.80	+3.60
MedGemma 4B	+0.60	−3.70	−2.20	+6.70

E.2 Cross-model transfer and model-specific routing

Table 15 evaluates each frozen router on all four target LVLMs. Because every entry is measured relative to the target model's own best fixed intervention, a positive value indicates that the transferred router contributes more than merely choosing a generally effective action.

The main (and expected) pattern is diagonal dominance: for every target LVLM, the router trained on that same LVLM gives the largest improvement. The matched routers exceed the corresponding best fixed interventions by 2.3, 2.1, 6.8, and 6.7 adequacy pp for Patho-R1 3B, Patho-R1 7B, Quilt-LLaVA 7B, and MedGemma 4B, respectively. The mean matched-model improvement is therefore approximately 4.5 pp, compared with only 1.4 pp when averaging the twelve off-diagonal transfers. This pattern supports fitting an LVLM-specific router rather than assuming that a single routing rule will be optimal for all perceptors.

We also note that the two Patho-R1 variants exhibit the strongest reciprocal transfer. The Patho-R1 3B router improves Patho-R1 7B by 1.7 pp, while the Patho-R1 7B router improves Patho-R1 3B by 1.8 pp, close to their matched-router gains of 2.3 and 2.1 pp. This suggests that the two models share part of their task–confidence treatment structure, plausibly because they belong to the same model family. Both Patho-R1 routers also transfer positively to Quilt-LLaVA and MedGemma, although they remain below the respective target-specific routers.

Transfer is less reliable for the more model-specific policies. The Quilt-LLaVA router performs strongly on Quilt-LLaVA itself (+6.8 pp) but falls below the best fixed intervention on Patho-R1 3B

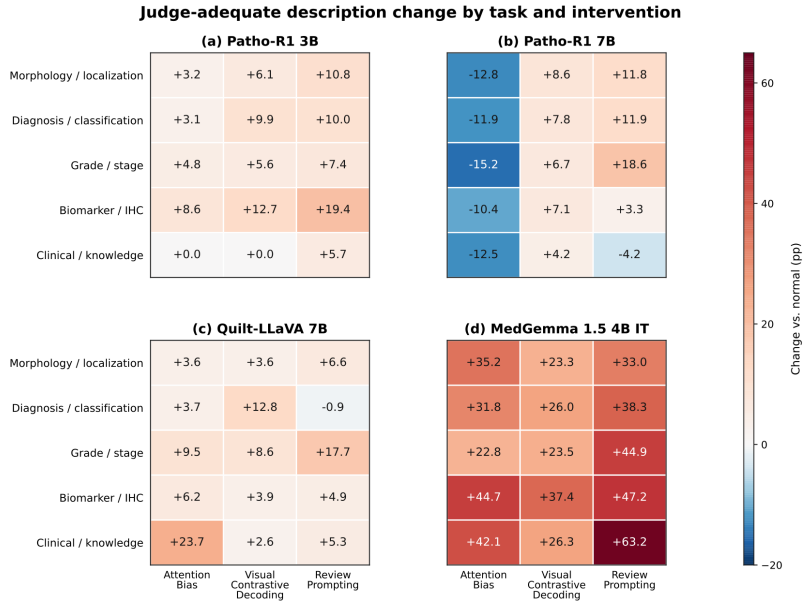


Figure 9: Task-specific changes in frozen-judge description adequacy for each intervention and LVLM. The preferred intervention is not stable across models or task types, motivating task-conditioned selection.

(-0.6) and Patho-R1 7B (-1.3). Similarly, the MedGemma router provides $+6.7$ pp on MedGemma but transfers negatively to Patho-R1 7B (-3.7) and Quilt-LLaVA (-2.2). Transfer is consequently asymmetric: a router that transfers from model m_1 to m_2 need not transfer equally well in the opposite direction.

This model dependence is expected for three reasons. First, adequacy confidence is calibrated separately for each LVLM, so the same numerical value of c_i^0 need not identify the same type of failure across models. Second, the interventions interact with model-specific components, including visual-token attention, decoding logits, and prompt following. Third, each router estimates action utilities from the candidate descriptions produced by its source LVLM; its coefficients therefore encode that model's own treatment-response surface, not only the shared question taxonomy.

The transfer matrix nevertheless shows that some routing structure is shared. In particular, all transfers from either Patho-R1 router remain positive. This suggests a possible extension in which related LVLMs share a hierarchical or partially pooled router while retaining model-specific calibration and residual coefficients. For the present proof of concept, however, the consistent diagonal advantage and the negative entries under mismatched transfer justify the safer choice of fitting and freezing a separate router for each LVLM. Since the table does not include uncertainty intervals for the transfer differences, these patterns are descriptive and should not be treated as formal evidence that every matched router is significantly better than every transferred router.

E.3 From fitted utilities to policy regions

Fig. 4 shows which intervention has the highest predicted candidate adequacy for each LVLM, question type, and baseline adequacy confidence. We construct these regions directly from the fitted logistic regressions.

Following Equation 3, let $\mathbf{e}_{\mathcal{T}}(t) \in \{0,1\}^{|\mathcal{T}|}$ denote the one-hot encoding of question type t. For baseline confidence c^0, the feature vector is

$$x(c^0, t) = \left[1,\ c^0,\ \mathbf{e}_{\mathcal{T}}(t)^\top,\ c^0 \mathbf{e}_{\mathcal{T}}(t)^\top\right]^\top \in \mathbb{R}^{2|\mathcal{T}|+2}. \tag{21}$$

Thus $x_i = x(c_i^0, t_i)$, and the dimension is 12 for the five question types used in this study. The first entry is constant, with corresponding unregularized intercept coefficient $w_{m,a,0}$.

For LVLM m and intervention a, write the fitted utility as a function of confidence and question type:

$$u_{m,a}(c^0, t) = \sigma\big(w_{m,a}^\top x(c^0, t)\big)\,. \tag{22}$$

This is the predicted probability that intervention a produces an adequate candidate. It is the same quantity as $u_{i,a} = u_{m,a}(c_i^0, t_i)$ in Equation 4.

For a fixed question type t, the logit is linear in confidence:

$$z_{m,a}(c^0, t) = w_{m,a}^\top x(c^0, t) = B_{m,a,t} + S_{m,a,t} c^0, \tag{23}$$

where $B_{m,a,t} = w_{m,a,0} + w_{m,a,\mathrm{task},t}$ and $S_{m,a,t} = w_{m,a,c} + w_{m,a,\mathrm{int},t}$ are the effective task-specific intercept and confidence slope defined in Equation 19. The interaction features therefore allow each intervention's confidence slope to differ across question types.

For an instance below the gate, the router selects

$$a_i^* = \arg\max_{a \in \mathcal{A}} u_{m,a}(c_i^0, t_i) = \arg\max_{a \in \mathcal{A}} z_{m,a}(c_i^0, t_i), \qquad c_i^0 < \tau. \tag{24}$$

The two maximizations agree because the sigmoid is strictly increasing. Exact ties are resolved in the fixed order ATTNB, VCD, then REVPROMPT.

To find where the preferred intervention changes, we compare the logit lines for each pair of interventions a and b. Their intersection satisfies

$$B_{m,a,t} + S_{m,a,t} c^0 = B_{m,b,t} + S_{m,b,t} c^0.$$

When the slopes differ, solving for confidence gives

$$c_{m,a,b,t}^* = \frac{B_{m,b,t} - B_{m,a,t}}{S_{m,a,t} - S_{m,b,t}}. \tag{25}$$

A crossing becomes a displayed action boundary only if it lies within the plotted confidence range and the maximizing intervention changes across it. Crossings below a third intervention's logit are omitted. Equal slopes produce no isolated crossing. If the intercepts also agree, the fixed tie-breaking rule applies.

The colored regions show the preferred intervention across $c^0 \in [0, 1]$, but intervention is executed only below the shared gate $\tau = 0.60$. The dashed gate therefore marks the decision to intervene, while the task-specific color boundaries mark which intervention to select. Candidate acceptance is a subsequent decision based on rescored confidence. These regions describe fitted action preferences, not observed call frequencies or guaranteed improvements.

E.4 How the fitted router reflects the observed treatment effects

The policy outcomes above establish that conditional routing is useful, while the transfer matrix establishes that its decision rule is partly model-specific. We next inspect what each router has learned. The task-level treatment effects in Fig. 9 describe differences averaged over baseline confidence, and the confidence-stratified effects in Fig. 10 describe how repairs and damage vary across confidence ranges. Fig. 4 combines these two sources of heterogeneity by visualizing the action with the largest fitted adequacy probability for each model, task, and confidence value.

Thus, the results presented in Fig. 4 reconcile the two marginal analyses in Fig.s 9 and 10. The task-specific heatmap explains which actions tend to be useful for a particular model and question type, but it averages over confidence. Conversely, the confidence-stratified plots show where interventions repair or damage descriptions, averaged over task composition. The router combines both signals through the $c^0 \times \mathbf{e}(t)$ interaction in Equation 21.

For Patho-R1 3B, Review Prompting is both the best fixed action and the task-only choice for all five tasks. Accordingly, most deployment-relevant router regions also select Review Prompting. The router nevertheless switches to Visual Contrastive Decoding near the upper end of the routed confidence range for diagnosis/classification and grade/stage. This agrees with the treatment-effect plots: Review Prompting supplies the strongest general rescue effect, while the relative advantage of Visual Contrastive Decoding increases for some medium-confidence cases.

Patho-R1 7B illustrates why a task-average treatment effect is insufficient. Attention Bias has a negative marginal effect in the task heatmap, yet the confidence-stratified analysis shows that it can produce repairs in the lowest-confidence region before becoming harmful at higher confidence. The router consequently assigns Attention Bias only below approximately $c^0 = 0.17$–0.23, depending on the task. It then changes to Review Prompting for morphology, diagnosis, and grade questions, and to Visual Contrastive Decoding for biomarker and clinical questions. The router therefore isolates a narrow region in which an intervention is useful even though applying it uniformly would reduce average adequacy.

Quilt-LLaVA exhibits the greatest task–confidence heterogeneity. Its best fixed action is Visual Contrastive Decoding, but the task-only lookup selects Review Prompting for morphology and grade, Visual Contrastive Decoding for diagnosis, and Attention Bias for biomarker and clinical questions. The full router further changes action within several tasks. In the deployment-relevant region $c^0 < 0.60$, morphology switches from Review Prompting to Attention Bias near 0.55, while biomarker questions switch from Review Prompting to Attention Bias near 0.39. These decisions mirror both the strong task-specific effects in Fig. 9 and the confidence-dependent repair–damage trade-offs in Fig. 10.

For MedGemma, the task identity explains most of the fitted policy: Attention Bias is selected throughout the morphology region, whereas Review Prompting is selected for diagnosis, grade, and clinical questions. Biomarker questions provide the main confidence-conditioned exception, switching from Review Prompting to Attention Bias near $c^0 = 0.52$. This is consistent with the large positive treatment effects of multiple interventions for MedGemma and shows that routing remains useful even when several interventions improve over ordinary inference.

Taken together, the panels show that the fitted policy is not simply recovering the best fixed arm or memorizing a task lookup table. It uses task type to represent broad differences in intervention suitability and baseline confidence to restrict each intervention to the region in which its predicted adequacy is greatest. Several analytically predicted transitions fall to the right of the gate threshold. These regions describe the fitted logistic models but are never executed by the deployed policy, because descriptions with $c^0 \geq 0.60$ are retained without intervention.

E.5 What the policy-region figure does not show

Fig. 4 is generated directly from the frozen router coefficients, gate threshold, best-fixed action, and task-only lookup stored in the frozen policy package. It visualizes the action that the router *would propose* for each (c^0, t) pair. It is not a histogram of observed decisions and does not account for the empirical density of examples along the confidence axis.

The figure also stops before rescoring acceptance. After the displayed action generates a candidate, ROUTER+ACCEPT may still reject it and return the ordinary description. The frozen package alone does not contain the per-example ordinary and candidate rescored confidence scores needed to simulate that decision. The plotted boundaries also represent point estimates of the fitted coefficients; they should not be interpreted as confidence intervals for the transition locations.

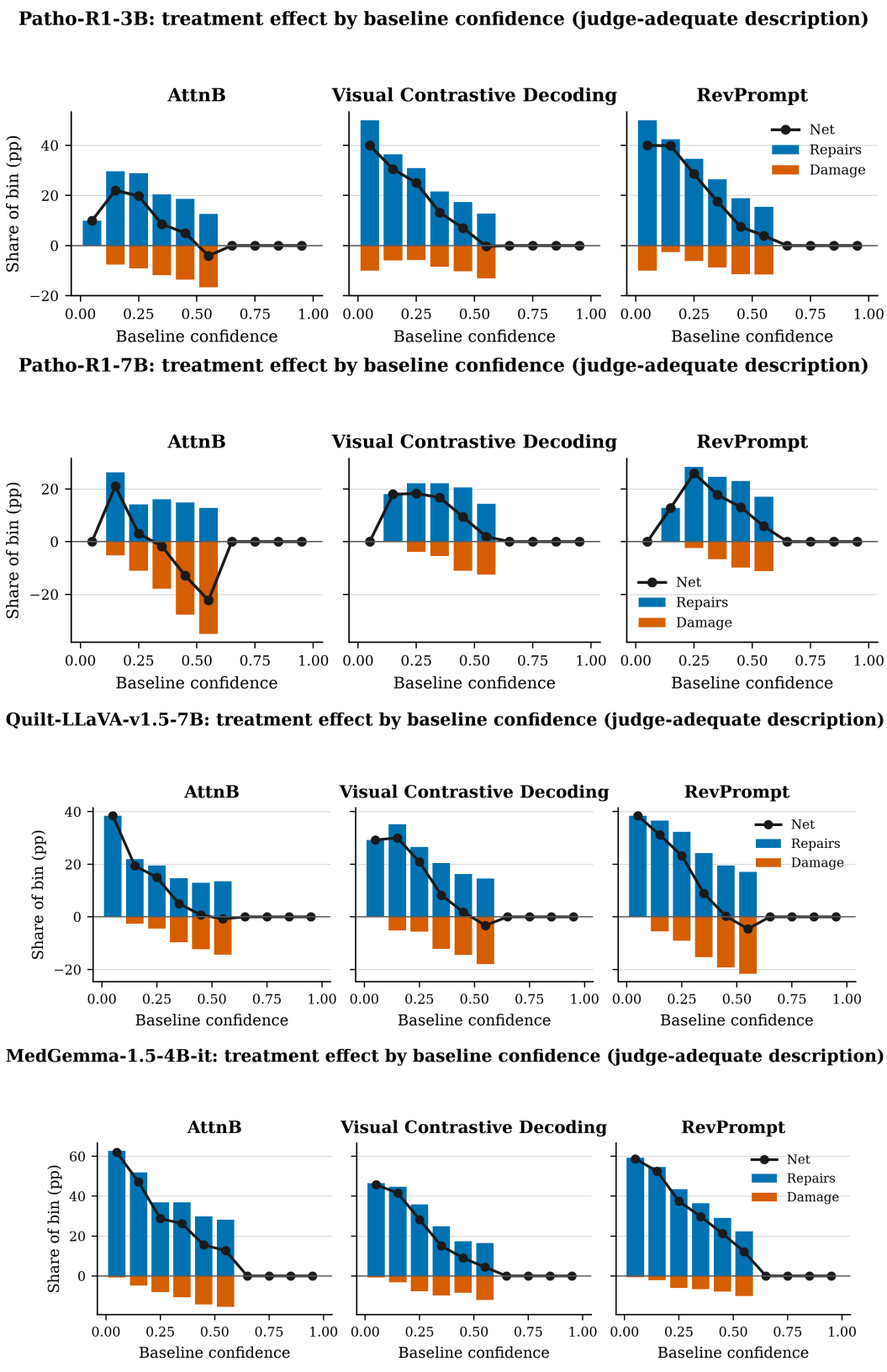


Figure 10: Repair, damage, and net treatment effects stratified by baseline adequacy confidence. Effects concentrate in the low- and medium-confidence regions and differ across interventions and LVLMs, motivating the confidence–task interaction.

F FROZEN EXPERIMENTAL PROTOCOL

Table 16: Prospectively fixed choices and validation-selected choices.

Component	Choice
Task taxonomy	Five question-only categories
Probe target	Reference-conditioned MedGemma-27B text-judge adequacy; no image
Probe prompts	Two variants per item; answer/reference hidden from the LVLM
Probe feature	Final layer, last in-answer token (`answer_last@-1`)
Probe model	Standardized feature; 32-unit GELU MLP; full-batch BCE
Probe fitting	Five-fold nested image grouping; overlap with `test_tiny` removed
Calibration	Positive-slope Platt scaling on disjoint image groups
ATTNB	Attention-only; $\alpha = 0.2$, layers 2 through final
VCD	$\alpha = 1.0$, $\beta = 0.1$, Gaussian blur radius 10
REVPROMPT	Frozen Qwen2.5-7B, two validated neutral procedures
Gate candidates	$\tau \in \{0.4, 0.5, 0.6, 0.8\}$
Acceptance candidates	Accept all; $c_a \geq c_0$; $c_a \geq c_0 + \delta$
Router	Candidate-adequacy logistic regression per intervention
Router features	Confidence, task one-hot, confidence–task interaction
Tie-break	ATTNB $<$ VCD $<$ REVPROMPT by cost
Decoding	Greedy, deterministic, seed 42
Uncertainty	Image-grouped paired bootstrap

G EXTENDED DISCUSSION AND LIMITATIONS

G.1 DISCUSSION

The proposed framework treats test-time intervention as a decision problem rather than a universally beneficial decoding modification. This distinction is especially important in agentic workflows. A fixed intervention can improve average description adequacy while still corrupting individual pieces of evidence that an agent later amplifies. Adequacy-confidence gating limits exposure to that risk, task-conditioned routing acknowledges that interventions have different inductive biases, and ordinary-rescoring based acceptance provides a conservative fallback. Because the probe predicts a judge verdict rather than answer correctness, downstream evaluation is not optional: it is the test of whether the local control signal matters to the agent.

The fixed-action experiments provide direct evidence for this risk. Review prompting, VCD, and attention biasing exchange rank across LVLMs and task strata, while the confidence-stratified repair and damage profiles show that even the same intervention changes character as baseline confidence increases. These patterns are inconsistent with a universal "always intervene" policy. Best fixed nevertheless remains strong, which is why the router claim is based on improvements over this validation-selected comparator rather than over normal inference alone.

The signal screen narrows the control problem further. Calibrated adequacy confidence is substantially more predictive than the attention-mass, image-dependence, blur-sensitivity, and answer-commitment alternatives in every completed LVLM. This is useful counter evidence: mechanistically appealing scores do not automatically make effective gates. At the same time, even the best observed confidence AUROC is imperfect. The gate therefore cannot be treated as an oracle. The acceptance rule and downstream paired evaluation remain necessary safeguards.

The small router is also a scientific instrument. Its purpose is not only to maximize accuracy, but to expose whether stable task–intervention relationships exist. The gains over task lookup show that task identity alone is insufficient, while the gains over confidence-only routing show that one confidence response curve cannot describe all tasks. The cross-model matrix further indicates partial but asymmetric portability: some source routers improve multiple target LVLMs, while negative cells argue against a universal router detached from the target model's inductive biases.

The WSI experiment tests a stronger form of generalization. The policy is developed from patch-level questions and frozen before insertion into agents that operate over pyramidal images and accumulated memories. Its highest macro-average accuracy across all six transfer blocks indicates that a

scalar adequacy interface and coarse task taxonomy can bridge model-level and system-level reliability in this setting. The result does not eliminate the possibility that richer state-aware confidence or memory-aware acceptance will be required for longer or more autonomous trajectories.

G.2 LIMITATIONS

Adequacy is assigned by a reference-conditioned text judge rather than through direct image inspection by pathologists. It therefore measures relevance, completeness, and consistency with the reference, but does not by itself establish image-grounded factuality. The blur-confidence-drop diagnostic in Table 8 provides counter-evidence to a purely answer-agreement explanation: after morphology is obliterated while the question and prompt remain fixed, confidence loss is positively associated with judge adequacy for all four LVLMs. However, the association is modest, and visual sensitivity is not equivalent to factual correspondence between a description and its patch. An image-aware audit by human pathologists stratified by question type, remain necessary to establish whether stronger adequacy scores reflect more faithful visual evidence rather than only reference-consistent language. But this is quite resource intensive and out of the scope of this study.

The five task categories deliberately compress pathology question diversity. Questions can require both morphology and clinical reasoning, but the main router assigns a single label. A multi-label or hierarchical taxonomy may improve routing at the cost of additional complexity and annotation uncertainty. This would be left for future studies.

The intervention set is not exhaustive. The study fixes one configuration for each intervention to limit researcher degrees of freedom; the resulting ranking may change under extensive per-model tuning. Conversely, this restriction makes differences easier to attribute to routing rather than hyperparameter search.

The intervention first acts only on a selected patch and cannot recover evidence from a diagnostically important region that neither agent initially visits. In PATHAGENT, however, the revised evidence can indirectly change later zoom and retrieval decisions; in the one-reflection SURPRISENAV setup, it chiefly changes adjudication. This asymmetry complicates a single causal interpretation and motivates reporting trajectories in future works, not only final answers.

Finally, evaluation is limited to four pathology LVLMs, two WSI agents, and the chosen benchmarks. Results may not generalize to other medical specialties, closed models, or systems in which the LVLM directly controls actions. Moreover, the simple controller is meant to act as scientific instrument to help us assess the utlity of description adequacy and conditional effect of common test time interventions in WSI agents and is not necessarily the optimal way to do interventions in WSI agentic systems. **The method is a research framework and is not intended for autonomous clinical use or deployment**.

H POLICY UTILIZATION AND EFFICIENCY

H.1 REALIZED POLICY UTILIZATION

Let N denote the number of ordinary LVLM perceptor calls, G the number for which $c^0 < \tau$, R_a the number routed to intervention $a \in \mathcal{A}$, $R = \sum_a R_a$, and A the number of candidates accepted after clean rescoring. We report

$$C_{\text{gate}} = \frac{G}{N}, \qquad C_{\text{accept}} = \frac{A}{R}, \qquad C_{\text{replace}} = \frac{A}{N}, \qquad S_a = \frac{R_a}{R}. \tag{26}$$

Here C_{gate} is the fraction of ordinary descriptions sent for rescue, C_{accept} is acceptance conditional on executing a routed intervention, and C_{replace} is the fraction of all perceptor calls whose description is ultimately replaced. The action shares S_a are also conditioned on routed calls and therefore sum to 100% within each row.

We also verified that the reported action shares sum to 100% up to rounding and that the number of routed calls does not exceed the number of gate-triggered calls.

Table 17: Realized utilization of the frozen ROUTER+ACCEPT policy. PathMMU rows use its frozen evaluation split. WSI rows pool call counts across SlideBench-TCGA, SlideBench-BCNB, and WSI-VQA for the indicated pipeline; percentages are computed from pooled numerators and denominators rather than averaging dataset-level percentages. Action shares are conditional on routed calls.

Setting	Perceptor LVLM	Calls (1k)(N)	Gate (%) (C_{gate})	Accept (%) (C_{accept})	Replace (%) (C_{replace})	S_{AttnB}	S_{VCD}	$S_{\text{RevPrompt}}$
PATHAGENT	Patho-R1 3B	41.19	72.21	70.81	51.13	0.0	7.3	92.7
	Patho-R1 7B	39.74	48.33	40.42	19.53	33.4	13.5	53.1
	Quilt-LLaVA 7B	68.78	65.81	67.56	44.46	16.0	35.0	49.0
	MedGemma 4B	43.54	35.38	80.91	28.63	31.3	0.0	68.7
SURPRISENAV	Patho-R1 3B	47.86	69.40	68.20	47.33	0.0	6.5	93.5
	Patho-R1 7B	45.92	51.60	43.80	22.60	32.9	13.5	53.6
	Quilt-LLaVA 7B	62.31	63.70	64.10	40.83	16.4	30.0	53.6
	MedGemma 4B	49.27	39.80	78.60	31.28	36.3	0.0	63.7

H.2 COMPONENT AND RUNTIME ACCOUNTING

The utilization statistics above also determine the number of additional model operations. Let $K_{\text{RP}} = 2$ be the frozen number of review procedures evaluated whenever REVPROMPT is selected. Relative to the ordinary description, the realized additional operation counts are

$$L_{\text{gen}} = R_{\text{AttnB}} + R_{\text{VCD}} + K_{\text{RP}} R_{\text{RevPrompt}}, \tag{27}$$

$$L_{\text{rescore}} = R_{\text{AttnB}} + R_{\text{VCD}} + K_{\text{RP}} R_{\text{RevPrompt}}, \tag{28}$$

$$L_{\text{blur}} = R_{\text{VCD}}, \tag{29}$$

$$M_{\text{external}} = R_{\text{RevPrompt}}. \tag{30}$$

Thus ATTNB adds one candidate generation and one clean rescoring; VCD adds one contrastive candidate generation, one blurred-image decoding branch, and one clean rescoring; and REVPROMPT makes one text-only meta-model call followed by two candidate generations and two clean rescorings. Rejected candidates remain in this accounting. Question-type labels are cached before agent inference and are reported separately from online meta-model calls.

All runtime measurements use the same NVIDIA GH200 configuration, software image, decoding limit, and agent settings as the accuracy experiments. We synchronize CUDA before and after every timed WSI question, exclude 3 warm-up questions, and report median and 95th-percentile end-to-end latency. Peak memory is the maximum allocated HBM after resetting PyTorch peak-memory statistics at the start of each question. GPU time sums wall-clock occupancy over every GPU used by the perceptor and local meta-model.

For policy π, the cost of a correctly repaired example is

$$\text{RepairCost}(\pi) = \frac{60\,(T_\pi - T_0)}{N^{\pi}_{W\to R}} \quad \text{GPU-minutes per wrong-to-right correction,} \tag{31}$$

where T_π and T_0 are GPU-hours per 100 questions for policy π and ordinary inference, respectively, and $N^{\pi}_{W\to R}$ is the number of wrong-to-right transitions per 100 questions. Repair cost is undefined when $N^{\pi}_{W\to R} = 0$.

Table 18: LVLM work is reported as mean additional candidate generations, clean rescorings, and VCD blurred-image branches per WSI question (Gen./Rep./Blur). Latency is median [95th percentile]. GPU time is reported per 100 questions. Paired wrong-to-right and right-to-wrong transitions are also normalized per 100 questions. Repair cost is incremental GPU-minutes over ordinary inference per wrong-to-right correction.

Pipeline	Perceptor LVLM	Policy	Gen./Rep./Blur	Meta calls/q.	Latency/q. (s)	Peak HBM (GB)	GPU-h/100 q.	W→R / R→W	$\mathrm{RepairCost}(\pi)$
PathAgent	Patho-R1 3B	Ordinary	0/0/0	0	42 [65]	22.4	1.32	–	–
		Only-AttnB	2.81/2.81/0	0	55 [80]	23.1	1.69	5.3/3.5	4.2
		Only-VCD	2.81/2.81/2.81	0	66 [95]	26.4	2.01	5.4/3.5	7.7
		Only-RevPrompt	5.61/5.61/0	2.81	79 [114]	39.6	2.41	3.2/5.1	20.4
		Router+Accept	5.41/5.41/0.20	2.60	78 [113]	39.6	2.38	5.2/1.5	12.2
	Patho-R1 7B	Ordinary	0/0/0	0	54 [84]	31.2	1.70	–	–
		Only-AttnB	1.81/1.81/0	0	66 [96]	31.9	2.02	8.7/3.5	2.2
		Only-VCD	1.81/1.81/1.81	0	75 [109]	36.8	2.29	7.5/3.5	4.7
		Only-RevPrompt	3.62/3.62/0	1.81	84 [122]	48.1	2.57	8.0/3.2	6.5
		Router+Accept	2.77/2.77/0.24	0.96	77 [112]	48.1	2.35	8.3/1.5	4.7
	Quilt-LLaVA 7B	Ordinary	0/0/0	0	75 [116]	32.6	2.35	–	–
		Only-AttnB	4.27/4.27/0	0	108 [157]	33.3	3.30	4.4/4.0	13.0
		Only-VCD	4.27/4.27/4.27	0	133 [193]	38.5	4.06	6.4/4.0	16.0
		Only-RevPrompt	8.54/8.54/0	4.27	160 [233]	49.3	4.90	4.6/4.0	33.3
		Router+Accept	6.36/6.36/1.49	2.09	142 [207]	49.3	4.35	5.5/2.0	21.8
	MedGemma 1.5 4B	Ordinary	0/0/0	0	47 [73]	25.2	1.48	–	–
		Only-AttnB	1.45/1.45/0	0	55 [80]	25.9	1.68	4.6/3.0	2.6
		Only-VCD	1.45/1.45/1.45	0	61 [88]	29.8	1.86	4.2/3.0	5.4
		Only-RevPrompt	2.91/2.91/0	1.45	68 [98]	42.1	2.07	3.8/3.0	9.3
		Router+Accept	2.45/2.45/0	1.00	64 [92]	42.1	1.95	4.3/1.3	6.6
SurpriseNav	Patho-R1 3B	Ordinary	0/0/0	0	38 [59]	21.8	1.19	–	–
		Only-AttnB	3.13/3.13/0	0	53 [77]	22.5	1.62	4.0/3.0	6.5
		Only-VCD	3.13/3.13/3.13	0	65 [94]	25.8	1.98	4.1/3.0	11.6
		Only-RevPrompt	6.27/6.27/0	3.13	79 [115]	39.0	2.42	3.5/3.0	21.1
		Router+Accept	6.06/6.06/0.20	2.93	78 [113]	39.0	2.39	3.0/1.3	24.0
	Patho-R1 7B	Ordinary	0/0/0	0	50 [78]	30.6	1.57	–	–
		Only-AttnB	2.24/2.24/0	0	65 [94]	31.3	1.98	6.7/3.0	3.7
		Only-VCD	2.24/2.24/2.24	0	76 [110]	36.2	2.32	5.4/3.0	8.3
		Only-RevPrompt	4.47/4.47/0	2.24	87 [127]	47.5	2.67	6.1/3.0	10.8
		Router+Accept	3.43/3.43/0.30	1.20	78 [114]	47.5	2.39	5.5/1.2	8.9
	Quilt-LLaVA 7B	Ordinary	0/0/0	0	65 [101]	32.0	2.04	–	–
		Only-AttnB	3.75/3.75/0	0	95 [137]	32.7	2.89	4.6/3.0	11.1
		Only-VCD	3.75/3.75/3.75	0	117 [170]	37.9	3.57	3.6/3.0	25.5
		Only-RevPrompt	7.49/7.49/0	3.75	142 [205]	48.7	4.33	3.8/3.0	36.2
		Router+Accept	5.75/5.75/1.12	2.01	127 [184]	48.7	3.87	4.1/1.5	26.8
	MedGemma 1.5 4B	Ordinary	0/0/0	0	43 [67]	24.6	1.35	–	–
		Only-AttnB	1.85/1.85/0	0	53 [77]	25.3	1.62	2.7/3.5	6.0
		Only-VCD	1.85/1.85/1.85	0	61 [88]	29.2	1.85	4.5/3.0	6.7
		Only-RevPrompt	3.70/3.70/0	1.85	69 [101]	41.5	2.12	4.0/3.0	11.6
		Router+Accept	3.03/3.03/0	1.18	63 [92]	41.5	1.94	4.1/1.2	8.6

I Implementation Controls and Compute

All components are loaded from frozen checkpoints in an Apptainer image. The router cache uses BF16 and is sharded to make the evals quicker. Agent jobs use one GH200 and ordinarily run two shards.

The complete WSI evaluation crosses four perceptor LVLMs with three datasets, two agents, and multiple correction policies. Moreover, a routed correction can add candidate generation and clean rescoring on top of an ordinary perceptor call. Running the larger inference budgets used in the original agent studies across this full evaluation grid was therefore not feasible within our compute allocation. We capped both agents at seven ordinary perceptor calls per question so that every job and experiment could run on one GH200 within the total compute limit allocated to our lab. This tighter budget restricts the amount of slide evidence that an agent can acquire and is the principal reason that our ordinary baseline accuracies are lower than the originally published results; these baselines should therefore not be read as exact reproductions of those absolute scores. The cap was nevertheless identical for ordinary inference and every fixed or routed policy within each dataset–agent–LVLM comparison, preserving the matched comparison needed to estimate policy gains. So we believe this limitation does not influence the conclusions of this study. Correction policies add model operations by design, but they receive no additional navigation steps or ordinary evidence acquisitions; their additional generations and rescorings are reported separately in Table 18.

We did not report confidence intervals for the WSI policy contrasts. A paired bootstrap over the retained final answers would quantify finite-benchmark uncertainty conditional on the deterministic trajectory realized for each question and policy, but it would not test the stability of an adaptive agent to changes earlier in its execution which is what is relevant to our study. Greedy decoding removes token sampling but does not by itself guarantee bitwise-identical GPU inference. Small floating-point differences can therefore alter an argmax when token logits are nearly tied, or cross a confidence gate, acceptance margin, or retrieval-ranking boundary. In an autoregressive agent these local changes can be amplified. An intervention can change the evidence memory and downstream reasoning and, for PathAgent, subsequent patch acquisition; resampling cached terminal outcomes cannot replay how such perturbations propagate through later state-dependent decisions. That broader robustness analysis would require running repeated end-to-end trajectories for every question-policy combination, and hierarchical analysis over trajectory replicates and slides. Repeating the full dataset-agent-LVLM-policy grid was not feasible within the compute limit allocated to our lab. However, we believe that the point estimates still provide useful evidence to make our case regarding the utility of the adequacy based signal.