MEDINSIGHT: ACTIVE CROSS-IMAGE EVIDENCE ACQUISITION FOR CLINICAL DIAGNOSIS

Anonymous authors
Paper under double-blind review

ABSTRACT

Clinical diagnosis from heterogeneous medical images requires finding and integrating visual evidence across multiple images. Existing methods lack the ability to actively acquire and align distributed visual evidence, often leading to missed subtle findings or incorrect cross-image relationships, thereby degrading overall performance. To address these challenges, we propose MedInSight, an end-to-end agentic framework for active cross-image clinical reasoning. Central to MedInSight is Cross-image Evidence Routing (CER), which leverages predictive uncertainty in per-image grounding and spatial uncertainty across images for targeted cross-image inspection. To optimize these visual interactions, we further propose Cross-horizon Credit Assignment (CCA), which combines short-horizon action-level credit magnitude with long-horizon stage-level direction for fine-grained credit assignment beyond uniform outcome-based advantages. Together, CER and CCA couple targeted cross-image evidence acquisition with fine-grained policy optimization, enabling progressive integration of distributed visual evidence for clinical diagnosis. We instill these capabilities through a two-stage training pipeline consisting of supervised fine-tuning (SFT) followed by reinforcement learning (RL). Extensive experiments demonstrate that MedInSight achieves state-of-the-art performance across medical diagnosis and reasoning benchmarks, improving its base model, Qwen3-VL-8B, by 22.96 points on average with consistent gains on out-of-domain datasets. Codes and checkpoints will be released.

1 INTRODUCTION

Remarkable advances have been made by multimodal large language models (MLLMs) in recent years. Excessive studies have leveraged them on various tasks in medical imaging analysis, such as visual question answering (VQA), radiology report generation, visual perception (grounding and segmentation), and clinical diagnosis. Furthermore, generalist medical systems, such as MedGemma (Sellergren et al., 2026), Lingshu (Xu et al., 2025), HealthGPT (Lin et al., 2025), HuatuoGPT-V (Chen et al., 2024) and MedXIAOHE (Shi et al., 2026), have begun to unlock the potential of MLLMs to support multiple medical tasks in a single unified framework.

Inspired by the success of reinforcement learning in incentivizing MLLMs to reason, recent studies (Lai et al., 2026; Sun et al., 2026; Huang et al., 2025) began to extend this paradigm to medical reasoning. Though the performance is promising, directly leveraging MLLMs has various issues. Importantly, relying primarily on pure-text reasoning is not explicitly grounded in the visual evidence. So, the model may attend to irrelevant regions and overlook important fine-grained evidence, thereby degrading the diagnostic performance.

To carefully integrate visual evidence into the reasoning process, recent studies (Jiang et al., 2026a; Li et al., 2026) introduce the thinking-with-images paradigm to enable models to reason over localized image regions. Though the performance is boosted, these approaches focus mainly on fine-grained reasoning *within* individual images. In contrast, clinical practice often requires interpreting findings across multiple heterogeneous images acquired across different modalities, views, and sequences. Diagnosis from such images requires grounding relevant regions, interpreting individual findings, establishing their cross-image relationships, and integrating visual evidence to progressively narrow the differential diagnosis. This is particularly challenging because lesion visibility can vary substantially across modalities, views, and sequences depending on its morphology, size,

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

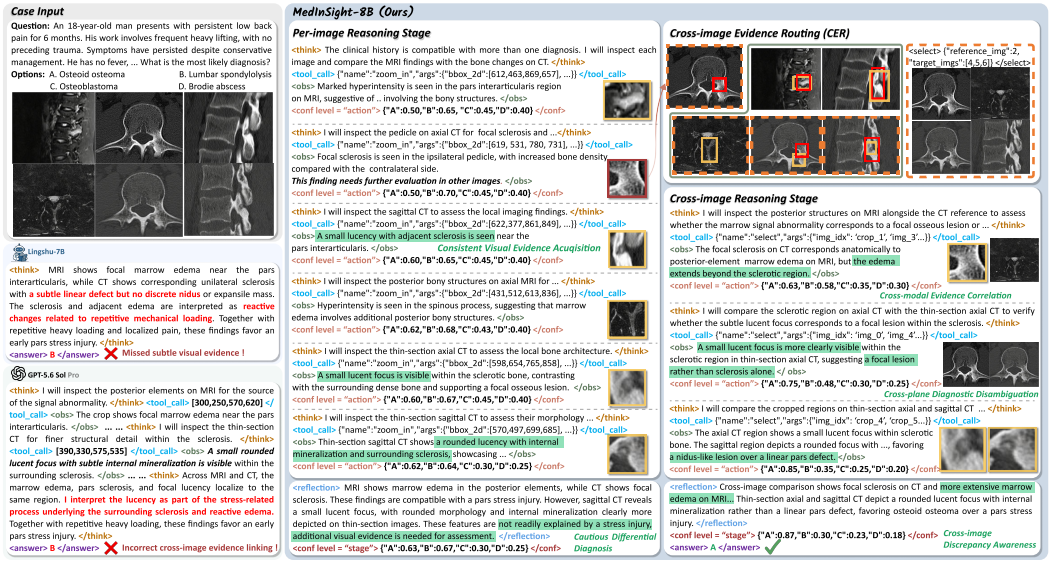


Figure 1: Pure-text reasoning and prompted thinking-with-image paradigms remain limited in localizing and linking diagnostic evidence across images. In contrast, our MedInSight leverages native grounding capability to actively route and relate visual evidence across images, enabling reliable cross-image reasoning for clinical diagnosis.

and anatomical location. As illustrated in Figure 1, the pure-text reasoning paradigm overlooks a subtle CT lesion, while simple interleaved visual-textual reasoning misinterprets its relationship with the MRI findings, leading to an incorrect diagnosis.

These failures motivate explicit cross-image reasoning to align related visual evidence and resolve diagnostic ambiguity through joint examination of relevant images and regions. Achieving this goal raises a fundamental research question: ***how to effectively acquire diagnostically informative evidence across heterogeneous medical images?*** Since lesion visibility varies across images, informative evidence is unevenly distributed, making it non-trivial to determine *which images or regions to examine* and *which resulting findings truly contribute to the correct diagnosis*. To address these challenges, we introduce **MedInSight**, an end-to-end agentic framework with two novel mechanisms. First, we propose **Cross-image Evidence Routing (CER)**, which leverages predictive uncertainty in per-image grounding and spatial uncertainty during cross-image regrounding to adaptively select images and regions for targeted evidence acquisition. This enables the model to revisit uncertain findings and reassess their cross-image relationships before advancing diagnostic reasoning. Second, outcome-based rewards assign the same trajectory-level advantage to all visual inspections, obscuring their different diagnostic contributions. To address this limitation, we propose **Cross-horizon Credit Assignment (CCA)**, which combines confidence transitions at two horizons for fine-grained credit assignment. Short-horizon transitions use the magnitude of confidence changes to weight action-level credit, while long-horizon transitions provide stage-level direction, determining whether each per-image or cross-image reasoning stage should be reinforced or suppressed. CCA combines these complementary signals to adjust action-level advantages without modifying the original outcome reward. Together, CER and CCA couple uncertainty-guided cross-image evidence acquisition with fine-grained policy optimization, enabling progressive visual evidence integration for clinical diagnosis.

Extensive experiments demonstrate that MedInSight achieves state-of-the-art (SOTA) performance consistently on both case-level diagnosis and multi-image medical reasoning, showcasing its generalizability and superior reasoning capabilities. Our main contributions are summarized as follows:

- We propose **MedInSight**, a novel end-to-end agentic framework that adaptively acquires and aligns localized evidence across heterogeneous medical images for clinical diagnosis.
- We design two novel, complementary mechanisms: **Cross-image Evidence Routing (CER)** for cross-image visual evidence acquisition and **Cross-horizon Credit Assignment (CCA)** for fine-grained advantage redistribution, jointly enabling the model to acquire, relate, and fine-grained policy optimization for clinical diagnosis.

- We establish new state-of-the-art results on multiple public medical VQA benchmarks, showcasing MedInSight's robust generalization and enhanced reasoning capabilities.

2 Related Works

Medical Reasoning with Multimodal Large Language Models. Multimodal large language models have demonstrated strong capabilities across diverse medical imaging modalities and tasks (Liu et al., 2025a; Zhao et al., 2025; Xu et al., 2025; Lin et al., 2025; Shi et al., 2026; Sellergren et al., 2026). Recent studies (Nath et al., 2025; Jiang et al., 2026b; Wang et al., 2026b; Zhu et al., 2026; Deria et al., 2026) further incorporate fine-grained visual evidence into the reasoning process, enabling medical MLLMs to acquire localized findings beyond global image descriptions. However, their training generally relies on terminal outcome or perception supervision, providing limited fine-grained credit for identifying which grounding operations contribute to reaching the correct answer. In contrast, MedInSight introduces cross-horizon credit assignment to differentiate intermediate inspection actions according to their estimated contributions to reaching the correct answer.

Multi-Image Medical Understanding. Recent medical MLLMs (Yang et al., 2025; Xu et al., 2026b; Chen et al., 2025; Zhang et al., 2026a; 2025; Liu et al., 2025b; Zhang et al., 2026b) have extended visual reasoning to multiple images, supporting longitudinal tracking, patient-clinician dialogue, and comparative understanding. Recent benchmarks (Wu et al., 2026; Yu et al., 2025b; Ning et al., 2026; Mu et al., 2025; Yao et al., 2026a; Xu et al., 2026c;a; Yue et al., 2026; Pan et al., 2026; Wang et al., 2026a) further evaluate multi-image diagnostic reasoning across diverse clinical settings, highlighting the importance of integrating distributed visual evidence across images. Despite these advances, existing methods largely rely on passive multi-image reasoning, where distributed visual evidence is processed as a whole without actively determining which findings warrant further inspection. This paradigm makes it difficult to capture fine-grained visual evidence and establish cross-image relationships across heterogeneous modalities, views, and sequences. MedInSight instead introduces native active perception, using iterative visual grounding and uncertainty-guided inspection to selectively revisit uncertain findings during cross-image reasoning.

3 Methodology

3.1 Method Overview

Accurate diagnosis from heterogeneous medical images requires an agent to determine not only which localized regions and image combinations to examine, but also which intermediate visual inspection actions to credit during training. To this end, we propose **MedInSight**, an agentic framework (Figure 2) that learns an adaptive policy for cross-image evidence acquisition by coupling native per-image grounding and uncertainty-guided joint inspection with cross-horizon credit assignment based on diagnostic feedback.

A case input $x = (c, \mathcal{I})$ contains clinical context c and images $\mathcal{I} = \{I^n\}_{n=1}^{N}$ from the same patient, spanning different modalities, anatomical planes, and acquisition sequences. Starting from this input, MedInSight first localizes and reasons over each image, producing per-image grounding proposals and visual findings that are integrated through a stage-level reflection to complete the per-image reasoning stage. The resulting grounding proposals are then passed to *Cross-image Evidence Routing* (CER), which leverages predictive and spatial uncertainty to select the images for subsequent cross-image reasoning. Specifically, MedInSight evaluates the predictive uncertainty of each per-image grounding proposal, selecting the most uncertain one as the anchor $A^\star$. Guided by this anchor, the model performs targeted re-grounding across all remaining images to extract their anatomically corresponding regions. Next, spatial uncertainty is computed between the anchor-conditioned bounding box proposal and the original per-image proposal within the same target image. CER then selects target images whose spatial uncertainty exceeds a predefined threshold and passes them, together with the anchor image, to the subsequent cross-image reasoning stage. During cross-image reasoning, MedInSight adaptively combines the selected whole images and regions specified by their existing per-image proposals for joint inspection. Each inspection produces a new observation and confidence readout; the observations are added to the reasoning history and integrated through a stage-level reflection. If the accumulated evidence remains insufficient, the selected images serve as

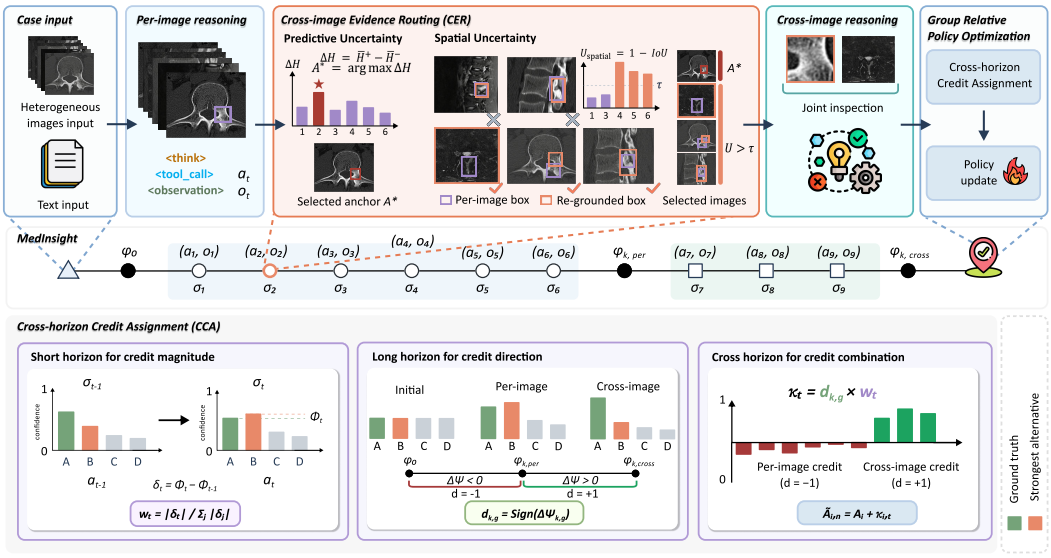


Figure 2: **Overview of MedInSight.** MedInSight employs CER to adaptively explore cross-image evidence based on the predictive and spatial uncertainty. CCA assigns fine-grained credit to visual inspection actions using short-horizon signals for action-level magnitude and long-horizon signals for stage-level direction. Together, CER and CCA enable targeted cross-image evidence acquisition and fine-grained policy optimization for clinical diagnosis.

the input to the next iteration, while all previously acquired findings are retained. This cycle of per-image reasoning, CER-based images selection, and cross-image reasoning repeats until MedInSight gathers sufficient evidence to formulate a final diagnosis, as formalized in Algorithm 1.

MedInSight is optimized via a two-stage training regime (Section 3.4): (i) cold-start supervised fine-tuning (SFT) to equip the base MLLM with foundational capabilities for visual action execution and evidence interpretation; and (ii) reinforcement learning (RL) to refine the visual inspection policy, with CER (Section 3.2) selecting images for further cross-image inspection and CCA (Section 3.3) providing fine-grained advantage adjustments.

3.2 Cross-image Evidence Routing

We design Cross-image Evidence Routing (CER) to select candidate images for cross-image reasoning, enabling efficient visual exploration across heterogeneous medical images. Given the case input $x = (c, \mathcal{I})$, MedInSight first performs per-image grounding to localize a suspected abnormal region B^n in each image I^n. Taking the per-image grounding results $\{(I^n, B^n)\}_{n=1}^{N}$ as input, CER sequentially applies two complementary uncertainty signals to produce an uncertainty-guided image subset $\mathcal{I}_{\text{select}}$ for subsequent joint cross-image inspection. Specifically, predictive uncertainty selects an anchor from the per-image grounding results for cross-image inspection, while spatial uncertainty identifies related images with localization evidence inconsistent with the anchor. These images are then selected for further cross-image inspection.

Predictive-uncertainty anchor selection. During per-image grounding, token-level entropy is monitored as predictive uncertainty during visual grounding command generation. Specifically, for the j-th token of the t-th grounding action a_t, predictive uncertainty is quantified as the entropy of the corresponding next-token probability distribution $H_{t,j} = -\sum_{v \in \mathcal{V}} q_{t,j}(v) \log q_{t,j}(v)$, where $q_{t,j}(v) = \pi_\theta\left(v \mid x, h_{<t}, a_{t,<j}\right)$. Here, $t \in \{1, \ldots, T\}$, $a_{t,<j}$ denotes the token prefix of the current grounding action before position j, $h_{<t}$ is the reasoning history before action a_t, and $\mathcal{V}$ is the model's vocabulary. Before the visual inspection action is started, a reference uncertainty value is first computed by averaging the token-level entropy. Once the tool call begins, entropy is continuously monitored over the generated bounding-box coordinate tokens, and their average is used to characterize uncertainty during localization. For each grounding proposal B^n, we denote the pre-trigger and post-trigger averages for grounding B^n by $\bar{H}_n^-$ and $\bar{H}_n^+$, respectively. The grounded proposal with

the largest entropy increase is directly selected as the anchor $A^\star = \underset{(I^n, B^n),\, n=1,\ldots,N}{\arg\max} \left(\bar{H}_n^+ - \bar{H}_n^-\right)$. Since bounding-box coordinates are generated as part of the grounding action, high entropy at these token positions indicates uncertainty about the location of the suspected abnormality. This signal serves as a dynamic trigger for selecting the most uncertain grounding proposal as the anchor, which then guides cross-image inspection through the spatial uncertainty introduced below.

Spatial-uncertainty image selection. Conditioned on the selected anchor $A^\star$, MedInSight regrounds the corresponding abnormality region in each remaining image I^m ($m \neq n^\star$), producing a cross-image proposal $B^{n^\star \to m}_{\text{cross}}$. Thus, each target image I^m is associated with two bounding-box proposals derived from distinct grounding paths: the per-image proposal B^m generated during initial per-image reasoning, and the anchor-conditioned cross-image proposal $B^{n^\star \to m}_{\text{cross}}$. Since both proposals are localized within the same target image, we define the spatial uncertainty $U^m_{\text{spatial}} = 1 - \text{IoU}\left(B^m, B^{n^\star \to m}_{\text{cross}}\right)$. A high U^m_{spatial} indicates that the suspected region B^m grounded in I^m is spatially inconsistent with the regrounded region $B^{n^\star \to m}_{\text{cross}}$ mapped from the anchor, revealing conflicting localization evidence across images. We retain all target images whose spatial uncertainty exceeds a predefined threshold τ_{spatial}. Together with $I^{n^\star}$, the selected image set is $\mathcal{I}_{\text{select}} = \left\{I^{n^\star}\right\} \cup \left\{ I^m \in \mathcal{I} \;\middle|\; m \neq n^\star,\ U^m_{\text{spatial}} > \tau_{\text{spatial}} \right\}$. The resulting image set $\mathcal{I}_{\text{select}}$ represents diverse yet targeted cross-image relationships driven by predictive and spatial uncertainty. MedInSight then jointly reasons over $\mathcal{I}_{\text{select}}$ during the following cross-image reasoning stage, with each inspection action a_t and its resulting observation o_t incorporated into the reasoning trajectory.

3.3 Cross-Horizon Credit Assignment

While MedInSight generates diverse visual inspection actions, identifying which intermediate actions genuinely contribute to the correct diagnosis remains challenging. An outcome-based reward assigns a constant trajectory-level advantage to all visual actions, failing to differentiate diagnostic evidence from irrelevant or misleading findings. To address this, we propose Cross-horizon Credit Assignment (CCA), which decouples credit into magnitude and direction across two complementary horizons. Specifically, short-horizon confidence shifts provide fine-grained action weights, yet their local directions are susceptible to transient fluctuations, which can inadvertently penalize beneficial inspection actions or reward harmful ones, ultimately degrading overall performance. For example, as shown in Figure 19, a temporary drop in confidence between consecutive steps can occur even when the agent is making steady progress toward the correct diagnosis. Accordingly, long-horizon transitions provide stage-level direction to guide action-level credit adjustment by capturing the cumulative diagnostic effect over a complete per-image or cross-image stage.

To operationalize this mechanism, we first define how diagnostic confidence is tracked along the reasoning trajectory. At each step t, the visual inspection action a_t is either a per-image grounding or a cross-image evidence combination, from which the agent generates an observation o_t. To quantify the diagnostic confidence change induced by a_t, the agent is prompted to report a confidence vector $\boldsymbol{\sigma}_t = \{\eta_t(\hat{y})\}_{\hat{y}\in\mathcal{Y}}$ over all candidate diagnoses $\mathcal{Y}$ after incorporating o_t into the reasoning history.

Short-horizon credit magnitude. To quantify diagnostic confidence from $\boldsymbol{\sigma}_t$, we define the truth-aligned margin as $\Phi_t = \eta_t(y) - \max_{\hat{y}\in\mathcal{Y}\setminus\{y\}} \eta_t(\hat{y})$, where y is the ground-truth answer. A larger Φ_t indicates stronger support for y over the strongest incorrect hypothesis. Before the first visual inspection, all candidate scores are initialized to the same neutral value, yielding $\Phi_0 = 0$.

Let $\mathcal{T} = \{1, \ldots, T\}$ index all visual inspection actions. For each $t \in \mathcal{T}$, the short-horizon diagnostic transition is $\delta^{\mathcal{S}}_t = \Phi_t - \Phi_{t-1}$, where Φ_t is computed after incorporating the observation o_t from action a_t. Thus, $\delta^{\mathcal{S}}_t$ serves as a proxy for the diagnostic contribution of each inspection, with Φ_0 providing the neutral reference for the first action. We normalize the transition magnitudes over the trajectory as $w^{\mathcal{S}}_t = |\delta^{\mathcal{S}}_t| / \sum_{j\in\mathcal{T}} |\delta^{\mathcal{S}}_j|$, providing a consistent credit scale across trajectories. If all transitions are zero, we set $w^{\mathcal{S}}_t = 0$ for all $t \in \mathcal{T}$, resulting in no credit adjustment.

Long-horizon credit direction. At the end of each reasoning stage $g \in \{\text{per}, \text{cross}\}$ in iteration k, the agent generates a reflection $r_{k,g}$ over the accumulated observations, followed by a stage-level confidence vector $\boldsymbol{\varphi}_{k,g} = \{\xi_{k,g}(\hat{y})\}_{\hat{y}\in\mathcal{Y}}$. Its truth-aligned margin is defined as $\Psi_{k,g} = \xi_{k,g}(y) -$

$\max_{\hat{y}\in\mathcal{Y}\setminus\{y\}} \xi_{k,g}(\hat{y})$. Consistent with the short-horizon formulation, setting the initial state as $\Psi_0 = \Phi_0$, we define the stage transitions for iteration k as $\Delta^{\mathcal{L}}_{k,\text{per}} = \Psi_{k,\text{per}} - \Psi_{k-1,\text{cross}}$ and $\Delta^{\mathcal{L}}_{k,\text{cross}} = \Psi_{k,\text{cross}} - \Psi_{k,\text{per}}$. The credit direction is defined as $d^{\mathcal{L}}_{k,g} = \text{Sign}(\Delta^{\mathcal{L}}_{k,g}) \in \{-1, 0, +1\}$, where a positive value indicates that the reasoning stage (k, g) improves the overall diagnosis.

Cross-horizon credit assignment. Finally, the two levels of credit signals are combined into a signed action-level coefficient $\kappa_t = d^{\mathcal{L}}_{k(t),g(t)} w^{\mathcal{S}}_t$, where $k(t)$ and $g(t)$ denote the iteration and reasoning stage to which inspection action a_t belongs, respectively.

Hence, the resulting κ_t is a signed action-level coefficient whose direction is determined by the long-horizon transition and magnitude by the normalized short-horizon transition. Positive and negative κ_t increase and decrease the advantage of the corresponding inspection action, respectively, while $\kappa_t = 0$ leaves it unchanged. During RL, κ_t adjusts the outcome-derived advantage without changing the terminal reward. Its token-level application is detailed in Section 3.4.

3.4 Training Strategy

Our training procedure consists of two stages: (i) cold-start Supervised Fine-Tuning (SFT) to equip the base MLLM with foundational capabilities for visual action execution and evidence interpretation; and (ii) reinforcement learning (RL) to further optimize the visual inspection policy, with CER guiding cross-image exploration and CCA providing fine-grained advantage adjustments.

Cold-Start Supervised Fine-Tuning. Direct RL optimization risks policy collapse, as the base model lacks structured visual inspection and evidence-integration behaviors for clinical diagnosis. To initialize these capabilities, we curate a cold-start dataset $\mathcal{D}_{\text{cold}}$ from the PubMed Central Open Access Subset, which contains expert-authored clinical case reports from real clinical examinations. Details of VQA construction, SFT trajectory generation, and quality control are provided in Appendix F.1. We minimize the average negative log-likelihood over all reasoning tokens:

$$\mathcal{L}_{\text{cold}} = \mathbb{E}_{(I,Q,y,R=\{(r_n,a_n,o_n)\}_{n=1}^{N})\sim\mathcal{D}_{\text{cold}}} \left(-\frac{1}{W} \sum_{n=1}^{N} \log p(r_n, a_n \mid I, Q, R_{<n}) \right). \tag{1}$$

where $W = \sum_{n=1}^{N} (|r_n| + |a_n|)$ is the total tokens of reasoning chain, r_n is textual thinking tokens.

Reinforcement Learning. We further optimize the initial policy π_θ via online Group Relative Policy Optimization (GRPO). For each training case $(c, \mathcal{I})$, a group of G trajectories $\{\chi_i\}_{i=1}^{G}$ is collected from the behavior policy $\pi_{\theta_{\text{old}}}$, with CER selecting images for further cross-image inspection. Each trajectory receives an outcome reward based on the correctness of its final diagnosis, $R_i = \mathbb{1}[\hat{y}_i = y]$, where $\hat{y}_i$ is the predicted answer in trajectory χ_i and y is the ground-truth answer.

The outcome-derived advantage A_i for the i-th trajectory is computed by normalizing the reward $A_i = \frac{R_i - \frac{1}{G}\sum_{j=1}^{G} R_j}{\text{std}(\{R_j\}_{j=1}^{G}) + \epsilon_{\text{adv}}}$, where $\epsilon_{\text{adv}} > 0$ ensures numerical stability. In vanilla GRPO, all policy-generated tokens in χ_i are assigned the same advantage A_i. To introduce action-level feedback, it suffices to apply additive shaping to the advantage by setting $\widetilde{A}_{i,n} \leftarrow A_i + \kappa_{i,t}$ for generated tokens associated with $a_{i,t}$ and $o_{i,t}$, where the adjustment coefficient $\kappa_{i,t}$ is defined in Section 3.3. Confidence readouts, stage-level reflections, and final-answer tokens retain A_i. Input tokens, grounding-returned visual content, and the initialization terms Φ_0 and Ψ_0 are masked from policy optimization. Thus, CCA modifies only token-level advantages, leaving the terminal reward and the GRPO framework unchanged. For the n-th token $z_{i,n}$ in trajectory χ_i, the surrogate objective is formulated as:

$$\mathcal{L}_{\text{RL}}(\theta) = \mathbb{E}_{\substack{(x,y)\sim\mathcal{D}_{\text{RL}} \\ \{\chi_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|x)}} \left[\frac{1}{G} \sum_{i=1}^{G} \frac{1}{Z_i} \sum_{n=1}^{|\chi_i|} M_{i,n} \ell_{i,n}(\theta) \right], \tag{2}$$

where $M_{i,n}$ is a binary mask that equals one for policy-generated tokens and zero otherwise, $Z_i = \sum_{n=1}^{N_i} M_{i,n}$ counts unmasked tokens. The per-token loss applies the shaped advantage $\widetilde{A}_{i,n}$:

$$\ell_{i,n}^{\text{clip}}(\theta) = -\min\left\{\rho_{i,n}(\theta)\widetilde{A}_{i,n}, \text{clip}\left(\rho_{i,n}(\theta), 1-\epsilon_{\text{low}}, 1+\epsilon_{\text{high}}\right)\widetilde{A}_{i,n}\right\}, \tag{3}$$

with clipping threshold $\epsilon > 0$ and $\rho_{i,n}(\theta) = \frac{\pi_\theta(z_{i,n}|h_{i,n})}{\pi_{\theta_{\text{old}}}(z_{i,n}|h_{i,n})}$ is the importance-sampling ratio. Here, $h_{i,n}$ denotes the recorded conditioning history before token $z_{i,n}$, including the case input and all preceding generated and tool-returned content. Both policies are evaluated under the same context. All advantages and CCA coefficients are computed after rollout and held fixed during policy updates.

4 Experiment

4.1 Experimental Setup

Benchmarks and Metrics. We evaluate MedInSight on six benchmarks spanning case-level diagnosis, clinical workflow reasoning, and multi-image understanding. We distinguish domains by data source: LiveClin, MMRareBench, and $\mathcal{D}_{\text{test}}$ are PMC-derived and treated as in-domain; MedThinkVQA, MedFrameQA, and MedXQA-Multi originate from Eurorad, YouTube medical videos, and medical examinations, respectively, and are treated as out-of-domain. We report final-answer accuracy for multi-image diagnosis on MedThinkVQA (Yao et al., 2026b), strict case accuracy for clinical workflow decisions on LiveClin (Wang et al., 2026a), and the rubric-based T1 diagnosis score on MMRareBench (Ning et al., 2026). For MedFrameQA (Yu et al., 2025b), we report average multiple-choice accuracy over questions with two to five video frames. MedXQA-Multi comprises the 419 questions with at least two images from the 2,000-question MedXpertQA-MM test split (Zuo et al., 2025), evaluated by exact-match accuracy. Our in-domain $\mathcal{D}_{\text{test}}$ comprises 311 VQA cases manually selected from the self-curated dataset (Section 3.4) after VQA construction, which are strictly withheld from SFT trajectory generation and RL training. For MedThinkVQA, LiveClin, and MedFrameQA, we retain published results where available and otherwise re-evaluate models under the corresponding protocols. We re-evaluate MedXQA-Multi and $\mathcal{D}_{\text{test}}$ using final-answer exact match. Details of $\mathcal{D}_{test}$ and other datasets are provided in Appendix D.2 and F.2.

Implementation Details. We use Qwen3-VL-8B (Bai et al., 2025a) with up to 768 visual tokens per image and a 32K-token context window. SFT, RL, and evaluation share the agent instruction template in Appendix E.2. For cold-start SFT, we train the model for three epochs with a learning rate of 1×10^{-5} and a global batch size of 64 on eight NVIDIA A100 GPUs. RL focuses on challenging queries for which best-of-N sampling produces no successful trajectory. Rollouts stop dynamically within a maximum of $T_{\max} = 32$ visual inspection actions. We use VERL (Sheng et al., 2025) on 16 NVIDIA A100 GPUs with DAPO's (Yu et al., 2025a) token-level loss aggregation and Clip-Higher, setting $\epsilon_{\text{low}} = 0.20$ and $\epsilon_{\text{high}} = 0.30$. The learning rate is fixed at 1×10^{-6}, with a global batch size of 128 and eight rollouts per group, without KL or entropy regularization.

4.2 Main Results

Table 1 shows that MedInSight achieves state-of-the-art performance among the evaluated general-purpose and medical-specific MLLMs on five of the six benchmarks. Compared with its Qwen3-VL-8B backbone, MedInSight improves the average score from 27.36% to 50.32%, yielding a gain of 22.96 points. On MedThinkVQA, MedInSight achieves 53.2% accuracy, substantially outperforming its base model, Qwen3-VL-8B (29.2%). Notably, it also surpasses the $4\times$ larger medical-specific Lingshu-32B (43.2%), demonstrating strong medical reasoning performance at a smaller model scale. Importantly, the gains generalize beyond the in-domain training source. MedInSight consistently improves over Qwen3-VL-8B across all three out-of-domain benchmarks, including gains of +24.0% on MedThinkVQA, +10.2% on MedFrameQA, and +32.2% on MedXQA-Multi, respectively. This indicates that the learned visual inspection actions and cross-image reasoning behaviors effectively transfer beyond the data source used for training. Furthermore, MedInSight remains highly competitive with proprietary models, ranking third on MedThinkVQA and MedFrameQA while achieving the highest score on $\mathcal{D}_{test}$. Notably, its average score of 50.32% surpasses Claude-Sonnet-4.5 (43.64%), Gemini 2.5 Flash (38.44%), and GPT-4o (36.93%), and approaches Claude 4.6 Opus (50.93%). Detailed benchmark scoring results are provided in Appendix B.1.

Table 1: **Main results on clinical diagnosis and reasoning benchmarks.** We evaluate MedInSight on six benchmarks spanning diverse clinical scenarios and imaging settings. AVG denotes the unweighted mean across the six benchmarks. **Bold** and underlined indicate the best and second-best results within each model category. Δ denotes the performance gain over Qwen3-VL-8B.

Models	Out-of-domain			In-domain			AVG
	MedThinkVQA↑	MedFrameQA↑	MedXQA-Multi↑	LiveClin↑	MMRareBench↑	$\mathcal{D}_{test}$-VQA↑	
Proprietary General-Purpose Models							
Claude 4.6 Opus	**57.2**	**56.8**	54.4	34.2	76.0	27.0	50.93
Claude-Sonnet-4.5	42.6	53.8	42.5	25.8	68.5	28.6	43.64
Gemini 3 Pro	55.3	48.2	76.9	30.1	76.7	34.7	**53.64**
Gemini 2.5 Pro	51.5	48.2	**79.5**	33.7	74.0	34.1	53.51
Gemini 2.5 Flash	41.8	54.8	27.5	19.0	62.2	25.4	38.44
GPT-5	48.6	52.8	68.7	35.5	75.9	32.8	52.38
GPT-4o	32.9	45.7	37.2	14.7	62.5	28.6	36.93
GPT-4o-mini	27.9	34.6	27.5	14.8	47.6	24.8	29.53
o3	45.0	50.2	65.2	**35.7**	**77.9**	**38.3**	52.05
Open-Source General-Purpose Models							
Qwen3.5-9B	**44.2**	**54.7**	**38.9**	**19.8**	51.8	**29.3**	**39.78**
Qwen3-VL-32B	32.9	49.5	32.2	17.7	**52.0**	24.8	34.85
InternVL3.5-8B	33.5	51.1	29.8	12.4	45.8	20.3	32.15
Qwen2.5-VL-32B	36.8	51.5	26.7	17.2	46.9	22.5	33.60
Qwen2.5-VL-7B	31.3	35.3	19.1	7.9	33.7	17.4	24.12
Medical-Specific Models							
Lingshu-32B	43.2	**54.5**	32.0	20.1	49.8	26.7	37.72
Lingshu-7B	36.9	42.2	27.0	11.1	37.9	22.2	29.54
MedGemma-27B-it	31.8	45.5	29.6	15.6	43.7	25.4	31.93
MedGemma-1.5-4B-it	29.6	41.8	23.0	7.8	23.5	19.3	24.16
MedGemma-4B-it	28.5	44.8	23.2	7.5	30.5	21.1	25.93
HuatuoGPT-VL-7B	30.6	45.6	23.6	9.2	22.8	22.2	25.67
Hulu-Med-14B	41.9	53.4	25.8	15.5	52.0	24.1	35.44
Hulu-Med-7B	36.3	48.5	27.2	10.3	42.2	22.8	31.22
Qwen3-VL-8B (Baseline)	29.2	44.0	22.7	10.4	39.3	18.6	27.36
MedInsight-8B (Ours)	**53.2**	54.2	**54.9**	**32.7**	**65.4**	**41.5**	**50.32**
Δ	*+24.0*	*+10.2*	*+32.2*	*+22.3*	*+26.1*	*+22.9*	*+22.96*

4.3 Ablation Study

Impact of training paradigms, CER, and CCA. Table 2 evaluates the contributions of CER and CCA on MedThinkVQA (MT), MedXQA-Multi (MX), and $\mathcal{D}_{\text{test}}$. **(i) Impact of evidence routing and credit assignment.** Our SFT model serves as the baseline without RL. Vanilla GRPO assigns outcome-derived trajectory-level advantages uniformly to all visual inspection actions. CER-only combines CER with vanilla GRPO, whereas CCA-only retains CCA but replaces CER with random image selection, preserving the cross-image inspection workflow. MedInSight combines both CER and CCA. Adding CER alone yields gains of 1.1-1.6 percentage points over vanilla GRPO. Building on CER, CCA provides a substantially larger gain of 6.2-6.8 points by adjusting action-level advantages according to their estimated diagnostic contributions. This suggests that *weighting actions by diagnostic correctness is more effective than assigning a uniform advantage across all action*s. Furthermore, the full model outperforms CCA-only with random image selection, showing that fine-grained credit assignment also benefits from targeted evidence routing. Together, these results highlight the complementary roles of CER and CCA. **(ii) CCA credit assignment.** With full CER retained, we ablate the two factors in the CCA coefficient $\kappa_t = d^{\mathcal{L}}_{k(t),g(t)} w^{\mathcal{S}}_t$. The long-horizon-only variant uses $\kappa_t = d^{\mathcal{L}}_{k(t),g(t)}$, retaining stage-level direction without action-specific magnitude weighting. The short-horizon-only variant uses $\kappa_t = w^{\mathcal{S}}_t$, retaining action-level magnitude without stage-level direction. Short-horizon credit consistently outperforms long-horizon credit across all three benchmarks, highlighting the importance of fine-grained credit redistribution. Notably, full CCA outperforms these two variants by 4.6-5.5 and 2.9-3.6 points, respectively. These consistent gains demonstrate the effectiveness of their complementary roles: the short-horizon signal determines the magnitude of action-level credit adjustment, while the long-horizon signal provides its direction based on stage-level diagnostic progress. **(iii) CER uncertainty signals.** With full CCA retained, we examine the two uncertainty signals by changing the anchor and target selection rules. The predictive-only variant retains predictive-uncertainty anchor selection but randomly selects the target images included in $\mathcal{I}_{\text{select}}$. The spatial-only variant randomly selects the anchor $A^\star$ and retains spatial uncertainty for target selection. Relative to full CER, predictive-only selection reduces accuracy by 1.7, 2.6, and 2.9 points on MT, MX, and $\mathcal{D}_{\text{test}}$, respectively; spatial-only selection reduces accuracy by 0.6, 1.4, and 1.3 points. The larger degradation from random target selection highlights the importance of spatial uncertainty for routing informative target images, while the consistent drop

Table 2: **Ablation study of training paradigms, CER, and CCA.** Accuracy (%) is reported on MedThinkVQA (MT), MedXQA-Multi (MX), and $\mathcal{D}_{\text{test}}$. Enabled signals are marked with ✓. CCA-only uses random image selection, CER-only uses the vanilla GRPO advantage.

Component	CER Uncertainty		CCA Credit		Reward	Accuracy ↑		
	Predictive	Spatial	Short	Long		MT	MX	$\mathcal{D}_{\text{test}}$
SFT						42.4	43.2	30.2
+ vanilla GRPO					✓	45.7	47.3	33.1
+ CER only	✓	✓			✓	46.8	48.7	34.7
+ CCA only			✓	✓	✓	51.1	51.6	37.6
Long-horizon credit only	✓	✓		✓	✓	48.6	49.4	36.3
Short-horizon credit only	✓	✓	✓		✓	50.3	51.6	37.9
CER with predictive only	✓		✓	✓	✓	51.5	52.3	38.6
CER with spatial only		✓	✓	✓	✓	52.6	53.5	40.2
MedInSight (CER + CCA)	✓	✓	✓	✓	✓	**53.2**	**54.9**	**41.5**

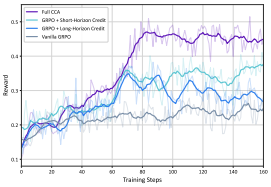


Figure 3: Training dynamics of rewards.

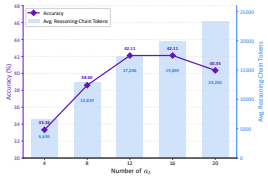


Figure 4: Test-time analysis of MedInSight.

from random anchor selection confirms the contribution of predictive uncertainty. Together, these results support the complementary roles of both uncertainty signals in CER. Comparisons with all-image inspection and alternative selection strategies are provided in Appendix B.2.

Analysis of training dynamics. As shown in Figure 3, full CCA exhibits more stable training dynamics and converges to a substantially higher reward plateau than vanilla GRPO and either short-horizon or long-horizon variant. Vanilla GRPO quickly saturates at a lower reward, suggesting that uniform trajectory-level credit provides limited supervision for further improving visual inspection behavior. Here, the short-horizon variant uses each local transition sign as its credit direction. While either single-horizon signal provides some improvement, combining short- and long-horizon signals progressively establishes a sustained reward advantage. These dynamics suggest that while fine-grained action-level credit is beneficial, relying on its local direction alone is insufficient; integrating it with the broader stage-level direction yields more effective policy optimization.

Test-time analysis. To examine whether MedInSight's gains simply arise from more reasoning computation, we control the number of input images by selecting all cases with exactly four images from $\mathcal{D}_{\text{test}}$, yielding a 57 case subset. We compare diagnostic accuracy against the mean reasoning-chain length. In Figure 4, each trajectory is set to perform a fixed number of visual inspection actions. Increasing the inspection budget initially improves accuracy, where accuracy climbs from 33.33% at 4 actions to 42.11% at 12 (+8.78 points). This indicates that deeper visual interaction initially helps uncover additional diagnostic evidence. Beyond 12 actions, however, even as reasoning-chain length grows by nearly 35%, accuracy saturates at 42.11% and eventually declines to 40.35% (-1.76 points). This trend suggests that excessive inspection may introduce redundant evidence and additional reasoning burden rather than improving diagnosis. *These results suggest that the gain depends on effective evidence acquisition rather than simply extending the trajectory*. Additional ablations and analyses of entropy dynamics and key hyperparameters are provided in Appendix B.3, with further analysis of CCA credit allocation in Appendix B.4.

5 Conclusion

We presented MedInSight, an end-to-end agentic framework for clinical diagnosis from heterogeneous medical images through native per-image grounding and adaptive cross-image inspection. MedInSight combines Cross-image Evidence Routing (CER) for uncertainty-guided cross-image evidence acquisition with Cross-horizon Credit Assignment (CCA) for fine-grained action-level credit assignment. Together, they enable targeted visual inspection and progressive integration of distributed evidence across images. Extensive experiments demonstrate strong performance and generalization across six benchmarks covering case-level diagnosis and multi-image medical reasoning. MedInSight also achieves competitive or superior performance relative to several proprietary models, demonstrating the effectiveness of active visual evidence acquisition without relying solely on increased model scale. These results highlight the value of active cross-image reasoning as a promising direction for clinical diagnosis. The limitations of MedInSight are discussed in Appendix G.

REFERENCES

Anthropic. Introducing Claude Sonnet 4.5, 2025. URL https://www.anthropic.com/news/claude-sonnet-4-5.

Anthropic. Introducing Claude Opus 4.6, 2026. URL https://www.anthropic.com/news/claude-opus-4-6.

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

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025b.

Junying Chen, Chi Gui, Ruyi Ouyang, Anningzhe Gao, Shunian Chen, Guiming Hardy Chen, Xidong Wang, Zhenyang Cai, Ke Ji, Xiang Wan, et al. Towards injecting medical visual knowledge into multimodal llms at scale. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pp. 7346–7370, 2024.

Zhen Chen, Yihang Fu, Gabriel Madera, Mauro Giuffre, Serina Applebaum, Hyunjae Kim, Hua Xu, and Qingyu Chen. From compound figures to composite understanding: Developing a multimodal llm from biomedical literature with medical multiple-image benchmarking and validation. *arXiv preprint arXiv:2511.22232*, 2025.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Ankan Deria, Komal Kumar, Adinath Madhavrao Dukre, Eran Segal, Salman Khan, and Imran Razzak. Medmo: Grounding and understanding multimodal large language model for medical images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5092–5103, June 2026.

Google DeepMind. Gemini 3 pro model card. Technical report, Google DeepMind, May 2026. URL https://deepmind.google/models/model-cards/gemini-3-pro/.

Xiaoke Huang, Juncheng Wu, Hui Liu, Xianfeng Tang, and Yuyin Zhou. Medvlthinker: Simple baselines for multimodal medical reasoning. *arXiv preprint arXiv:2508.02669*, 2025.

Songtao Jiang, Yuan Wang, Sibo Song, Tianxiang Hu, Chenyi Zhou, Bin Pu, Yan Zhang, Zhibo Yang, Yang Feng, Joey Tianyi Zhou, et al. Hulu-med: A transparent generalist model towards holistic medical vision-language understanding. *arXiv preprint arXiv:2510.08668*, 2025.

Yankai Jiang, Yujie Zhang, Peng Zhang, Wenjie Li, Yichen Li, Jintai Chen, Xiaoming Shi, and Shihui Zhen. Ophiuchus: Incentivizing tool-augmented "think with images" for joint medical segmentation, understanding and reasoning. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026a.

Zheng Jiang, Heng Guo, Chengyu Fang, Changchen Xiao, Xinyang Hu, Lifeng Sun, and Minfeng Xu. Medvr: Annotation-free medical visual reasoning via agentic reinforcement learning. In *The Fourteenth International Conference on Learning Representations*, 2026b.

Yuxiang Lai, Jike Zhong, Ming Li, Shitian Zhao, Yuheng Li, Konstantinos Psounis, and Xiaofeng Yang. Med-r1: Reinforcement learning for generalizable medical reasoning in vision-language models. *IEEE transactions on medical imaging*, 2026.

Wenjie Li, Yujie Zhang, Haoran Sun, Xingqi He, Hongcheng Gao, Chenglong Ma, Ming Hu, Guankun Wang, Shiyi Yao, Renhao Yang, Hongliang Ren, Lei Wang, Junjun He, and Yankai Jiang. Medscope: Incentivizing "think with videos" for clinical reasoning via coarse-to-fine tool calling. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

Tianwei Lin, Wenqiao Zhang, Sijing Li, Yuqian Yuan, Binhe Yu, Haoyuan Li, Wanggui He, Hao Jiang, Mengze Li, Xiaohui Song, et al. Healthgpt: A medical large vision-language model for unifying comprehension and generation via heterogeneous knowledge adaptation. *arXiv preprint arXiv:2502.09838*, 2025.

Anglin Liu, Rundong Xue, Xu R. Cao, Yifan Shen, Yi Lu, Xiang Li, Qianqian Chen, and Jintai Chen. MedSAM3: Delving into segment anything with medical concepts, 2025a. URL https://arxiv.org/abs/2511.19046.

Kang Liu, Zhuoqi Ma, Xiaolu Kang, Yunan Li, Kun Xie, Zhicheng Jiao, and Qiguang Miao. Enhanced contrastive learning with multi-view longitudinal data for chest x-ray report generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10348–10359. IEEE, 2025b.

Linjie Mu, Zhongzhen Huang, Shengqian Qin, Yakun Zhu, Shaoting Zhang, and Xiaofan Zhang. Mmxu: A multi-modal and multi-x-ray understanding dataset for disease progression. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 9785–9803, 2025.

Vishwesh Nath, Wenqi Li, Dong Yang, Andriy Myronenko, Mingxin Zheng, Yao Lu, Zhijian Liu, Hongxu Yin, Yee Man Law, Yucheng Tang, Pengfei Guo, Can Zhao, Ziyue Xu, Yufan He, Stephanie Harmon, Benjamin Simon, Greg Heinrich, Stephen Aylward, Marc Edgar, Michael Zephyr, Pavlo Molchanov, Baris Turkbey, Holger Roth, and Daguang Xu. VILA-M3: Enhancing vision-language models with medical expert knowledge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14788–14798, June 2025.

Junzhi Ning, Jiashi Lin, Yingying Fang, Wei Li, Jiyao Liu, Cheng Tang, Chenglong Ma, Wenhao Tang, Tianbin Li, Ziyan Huang, et al. Mmrarebench: A rare-disease multimodal and multi-image medical benchmark. *arXiv preprint arXiv:2604.10755*, 2026.

OpenAI. GPT-4o system card. Technical report, OpenAI, 2024a. URL https://openai.com/index/gpt-4o-system-card/.

OpenAI. GPT-4o mini: Advancing cost-efficient intelligence, 2024b. URL https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/.

OpenAI. GPT-5 system card. Technical report, OpenAI, 2025a. URL https://openai.com/index/gpt-5-system-card/.

OpenAI. OpenAI o3 and o4-mini system card. Technical report, OpenAI, 2025b. URL https://openai.com/index/o3-o4-mini-system-card/.

Jiazhen Pan, Weixiang Shen, Jun Li, Julian Canisius, Felix Bitzer, Paula RoĆ mĆ1/4ller, Jiancheng Yang, Virginie Kreutzinger, Daniel Rueckert, and Benedikt Wiestler. Ddx-trace: A benchmark for medical diagnostic trajectories in vlms. *arXiv preprint arXiv:2605.23629*, 2026.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Andrew Sellergren, Sahar Kazemzadeh, Tiam Jaroensri, Atilla Kiraly, Madeleine Traverse, Timo Kohlberger, Shawn Xu, Fayaz Jamil, Cían Hughes, Charles Lau, et al. Medgemma technical report. *arXiv preprint arXiv:2507.05201*, 2025.

Andrew Sellergren, Chufan Gao, Fereshteh Mahvar, Timo Kohlberger, Fayaz Jamil, Madeleine Traverse, Alberto Tono, Bashir Sadjad, Lin Yang, Charles Lau, et al. Medgemma 1.5 technical report. *arXiv preprint arXiv:2604.05081*, 2026.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. HybridFlow: A flexible and efficient RLHF framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, EuroSys '25, pp. 1279–1297. ACM, 2025. doi: 10.1145/3689031.3696075.

Baorong Shi, Bo Cui, Boyuan Jiang, Deli Yu, Fang Qian, Haihua Yang, Huichao Wang, Jiale Chen, Jianfei Pan, Jieqiong Cao, et al. Medxiaohe: A comprehensive recipe for building medical mllms. *arXiv preprint arXiv:2602.12705*, 2026.

Haoran Sun, Yankai Jiang, Wenjie Lou, Yujie Zhang, Wenjie Li, Lilong Wang, Mianxin Liu, Lei Liu, and Xiaosong Wang. Chiron-o1: Igniting multimodal large language models towards generalizable medical reasoning via mentor-intern collaborative search. *Advances in Neural Information Processing Systems*, 38:104180–104217, 2026.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Xidong Wang, Yue Shen, Junying Chen, Jian Wang, Jinjie Gu, Ping Zhang, Lei Liu, Wang Benyou, et al. Liveclin: A live clinical benchmark without leakage. In *International Conference on Learning Representations*, volume 2026, pp. 97981–98011, 2026a.

Ziyue Wang, Junde Wu, Linghan Cai, Chang Low, Xihong Yang, Qiaxuan Li, and Yueming Jin. Medagent-pro: Towards evidence-based multi-modal medical diagnosis via reasoning agentic workflow. In *International Conference on Learning Representations*, volume 2026, pp. 114360–114389, 2026b.

Zhilin Wu, Zhangkai Ni, Chengmei Yang, Longzhen Yang, Yihang Liu, Ying Wen, and Lianghua He. Lomevqa: A comprehensive benchmark for longitudinal medical vqa. *arXiv preprint arXiv:2607.27806*, 2026.

Dexuan Xu, Yuan Jiayin, Jianing Wang, Yanyuan Chen, Hanpin Wang, and Yu Huang. Beyond single view: A comprehensive benchmark for medical multimodal large language models on multi-image understanding. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 27376–27394, San Diego, California, United States, July 2026a. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.1263. URL https://aclanthology.org/2026.acl-long.1263/.

Jiao Xu, Junwei Liu, Jiangwei Lao, Qi Zhu, Yunpeng Zhao, Congyun Jin, Shinan Liu, Zhihong Lu, Lihe Zhang, Xin Chen, et al. Pulsemind: A multi-modal medical model for real-world clinical diagnosis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 11259–11268, 2026b.

Ronghao Xu, Zhen Huang, Yangbo Wei, Xiaoqian Zhou, Zikang Xu, Ting Liu, Zihang Jiang, and S. Kevin Zhou. Medatlas: Evaluating large language models for multi-round, multi-task medical reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026c.

Weiwen Xu, Hou Pong Chan, Long Li, Mahani Aljunied, Ruifeng Yuan, Jianyu Wang, Chenghao Xiao, Guizhen Chen, Chaoqun Liu, Zhaodonghui Li, et al. Lingshu: A generalist foundation model for unified multimodal medical understanding and reasoning. *arXiv preprint arXiv:2506.07044*, 2025.

Xikai Yang, Juzheng Miao, Yuchen Yuan, Jiaze Wang, Qi Dou, Jinpeng Li, and Pheng-Ann Heng. Medical large vision language models with multi-image visual ability. In *Proceedings of Medical Image Computing and Computer Assisted Intervention*, pp. 402–412. Springer Nature Switzerland, 2025.

Zonghai Yao, Benlu Wang, Yifan Zhang, Junda Wang, Iris Xia, Zhipeng Tang, Shuo Han, Feiyun Ouyang, Zhichao Yang, Arman Cohan, and Hong Yu. Medical thinking with multiple images. *arXiv preprint arXiv:2604.16506*, 2026a.

Zonghai Yao, Benlu Wang, Yifan Zhang, Junda Wang, Iris Xia, Zhipeng Tang, Shuo Han, Zhichao Yang, Arman Cohan, and Hong Yu. Medical thinking with multiple images. In *International Conference on Learning Representations*, volume 2026, pp. 101924–101988, 2026b.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale. In *Advances in Neural Information Processing Systems*, volume 38, pp. 113222–113244. Curran Associates, Inc., 2025a. doi: 10.52202/085713-3775.

Suhao Yu, Haojin Wang, Juncheng Wu, Luyang Luo, Jingshen Wang, Cihang Xie, Pranav Rajpurkar, Carl Yang, Yang Yang, Kang Wang, et al. Medframeqa: A multi-image medical vqa benchmark for clinical reasoning. *arXiv preprint arXiv:2505.16964*, 2025b.

Jingkun Yue, Siqi Zhang, Zinan Jia, Huihuan Xu, Zongbo Han, Xiaohong Liu, and Guangyu Wang. Medsg-bench: A benchmark for medical image sequences grounding. *Advances in Neural Information Processing Systems*, 38, 2026.

Tengfei Zhang, Ziheng Zhao, Xiaoman Zhang, Lisong Dai, Pengcheng Qiu, Ya Zhang, Yanfeng Wang, and Weidi Xie. A vision-language framework for comparative reasoning in radiology. *arXiv preprint arXiv:2606.06407*, 2026a.

Xi Zhang, Zaiqiao Meng, Jake Lever, and Edmond S. L. Ho. Libra: Leveraging temporal images for biomedical radiology analysis. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 17275–17303, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.888. URL https://aclanthology.org/2025.findings-acl.888/.

Yuting Zhang, Kaishen Yuan, Hao Lu, Yutao Yue, Jintai Chen, and Kaishun Wu. Medtvt-r1: A multimodal llm empowering medical reasoning and diagnosis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 35248–35259, 2026b.

Theodore Zhao, Yu Gu, Jianwei Yang, Naoto Usuyama, Ho Hin Lee, Sid Kiblawi, Tristan Naumann, Jianfeng Gao, Angela Crabtree, Jacob Abel, Christine Moung-Wen, Brian Piening, Carlo Bifulco, Mu Wei, Hoifung Poon, and Sheng Wang. A foundation model for joint segmentation, detection and recognition of biomedical objects across nine modalities. *Nature Methods*, 22(1):166–176, 2025. doi: 10.1038/s41592-024-02499-w.

Chunzheng Zhu, Yangfang Lin, Shen Chen, Yijun Wang, and Jianxin Lin. Medeyes: Learning dynamic visual focus for medical progressive diagnosis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 13916–13924, 2026. doi: 10.1609/aaai.v40i16.38401.

Yuxin Zuo, Shang Qu, Yifei Li, Zhang-Ren Chen, Xuekai Zhu, Ermo Hua, Kaiyan Zhang, Ning Ding, and Bowen Zhou. MedxpertQA: Benchmarking expert-level medical reasoning and understanding. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=IyVcxU0RKI.

CONTENTS

1 Introduction **1**

2 Related Works **3**

3 Methodology **3**

3.1 Method Overview 3

3.2 Cross-image Evidence Routing 4

3.3 Cross-Horizon Credit Assignment 5

3.4 Training Strategy 6

4 Experiment **7**

4.1 Experimental Setup 7

4.2 Main Results 7

4.3 Ablation Study 8

5 Conclusion **9**

A LLM Usage Disclosure **16**

B Supplementary Experimental Results **17**

B.1 Detailed Benchmark Results 17

B.2 Effectiveness of CER 19

B.3 Training Dynamics and Hyperparameter Analysis 19

B.4 CCA Credit Allocation Analysis 20

C Algorithm Workflow **23**

D Implementation Details **25**

D.1 Model and Training Configuration 25

D.2 Benchmark Details 25

D.3 Baselines and Evaluation Protocols 26

E Inference Settings and Prompts **27**

E.1 Visual Inspection Protocol 27

E.2 Prompt Templates 27

F Dataset Details **34**

F.1 Data Construction 34

F.2 $\mathcal{D}_{test}$-VQA Details 35

G Limitations and Future Work **37**

H Qualitative Case Studies **38**

I Ethics Statement **42**

J Reproducibility Statement **42**

A LLM USAGE DISCLOSURE

In accordance with the ICLR 2027 policies on LLM usage, we disclose how LLMs were used in this work. First, LLMs were used for writing assistance, including language polishing, grammar correction, and improving the overall readability of the manuscript. Second, LLMs were incorporated into parts of the data construction pipeline, where they are used for our data construction. All technical content, proofs, experiments, and analyses were conceived, implemented, and validated by the authors. Authors remain fully responsible for the correctness of the claims and results. No LLMs were used to generate research ideas, write code for experiments, or produce results. No confidential information was shared with LLMs, and no prompt injections or other inappropriate uses were involved. This disclosure aligns with the ICLR Code of Ethics: contributions of tools are acknowledged, while accountability and verification rest entirely with the human authors.

B SUPPLEMENTARY EXPERIMENTAL RESULTS

B.1 DETAILED BENCHMARK RESULTS

Detailed results for the individual benchmarks are presented in Figure 5, Figure 6, Figure 7 and Table 3.

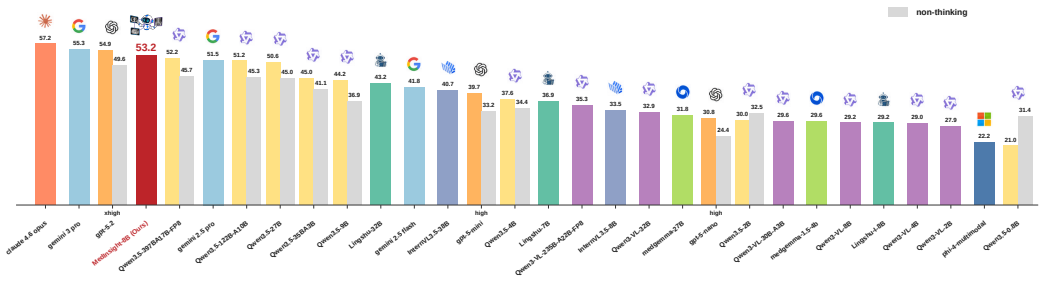


Figure 5: **Detailed diagnostic accuracy on MedThinkVQA.** Final-answer accuracy (%) on the 720-case test set. Paired bars compare thinking and matched non-thinking configurations when both are available; gray denotes non-thinking. Single bars show the available thinking configuration or, for models without one, the standard configuration. MedInSight-8B achieves 53.2%, above the displayed open-source alternatives.

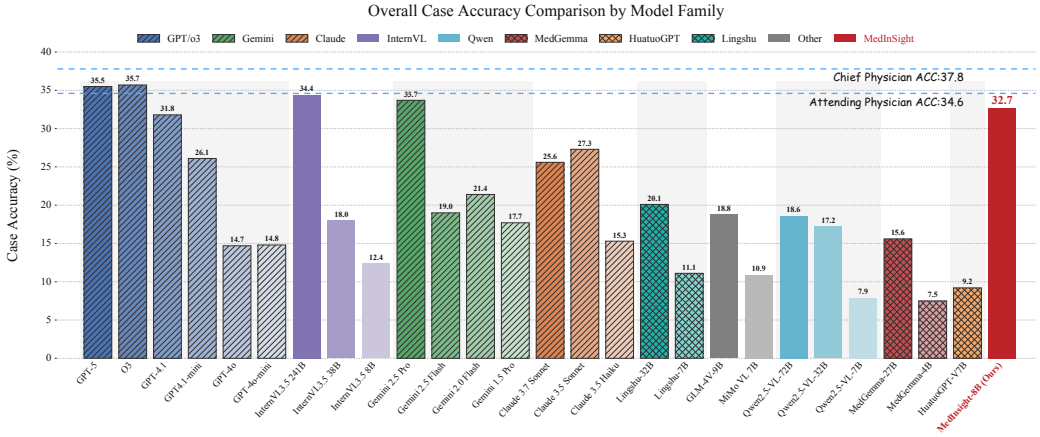


Figure 6: **Case-level clinical workflow accuracy on LiveClin.** A case is counted as correct only when all its questions are answered correctly. Colors and patterns distinguish model families; dashed lines show the physician reference levels reported by the benchmark. MedInSight-8B achieves 32.7% case accuracy.

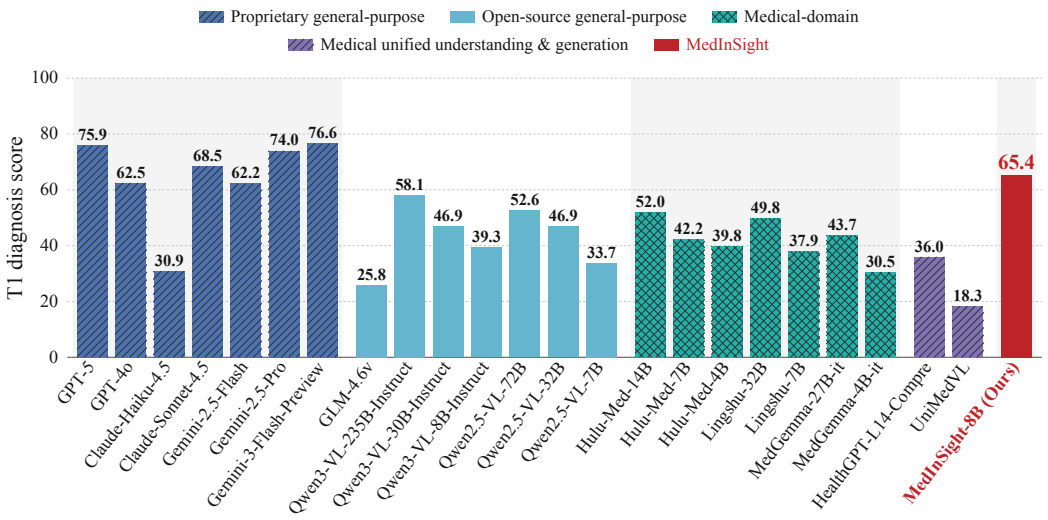


Figure 7: **Rare-disease diagnosis on MMRareBench.** Rubric-based T1 diagnosis scores on the 474-case test subset, on a 0-100 scale (higher is better). This metric assesses open-ended diagnostic answers rather than multiple-choice accuracy. MedInSight-8B scores 65.4, above the listed open-source general-purpose and medical-domain alternatives.

Table 3: **System-wise multi-frame understanding on MedFrameQA.** Multiple-choice accuracy (%) across the nine clinical-system categories, using the benchmark's category abbreviations. System scores aggregate the corresponding tasks; Avg reports the benchmark's overall task accuracy. MedInSight-8B achieves an overall accuracy of 54.19%, with performance varying across clinical systems.

Model	Accuracy per System									Avg
	CNS	RES	CIR	DIG	URI	REP	END	MSK	AUX	
Proprietary Reasoning Models										
o1	46.91	48.88	49.49	47.45	49.03	42.26	47.68	51.59	48.75	47.91
o3	47.81	52.00	50.00	48.48	50.71	45.02	51.84	54.90	50.41	50.18
o4-mini	46.03	49.78	48.74	48.63	51.85	43.62	52.44	53.38	50.82	49.40
Gemini-2.5-Flash	48.82	**58.26**	**57.21**	**50.25**	48.61	**55.81**	55.38	**60.21**	52.85	**54.75**
Claude-3.7-Sonnet	49.21	46.09	53.23	**50.25**	49.07	47.57	47.81	52.42	49.59	49.67
Open-Source Reasoning Models										
QvQ-72B-Preview	44.88	46.67	47.43	41.13	45.68	47.00	47.68	49.37	47.15	46.44
Proprietary Non-Reasoning Models										
GPT-4o	48.82	49.13	37.31	50.00	43.98	45.88	46.22	43.60	44.31	45.67
GPT-4o-mini	41.73	36.52	39.30	28.36	35.65	33.83	30.68	34.95	34.96	34.55
GPT-4-Turbo-V	45.28	46.09	42.79	49.75	43.06	48.63	49.80	45.16	46.75	46.69
Open-Source Non-Reasoning Models										
Qwen2.5-VL-72B	43.18	47.39	42.29	39.80	39.81	43.41	43.03	44.00	40.11	42.65
Open-Source Non-Reasoning Medical Finetuned Models										
MedGemma-27b-it	49.61	44.20	48.09	43.45	41.36	46.58	50.33	45.62	39.70	45.47
MedInsight-8B(Ours)	**51.97**	57.83	56.71	**50.25**	**54.17**	52.78	**55.78**	58.48	**54.47**	54.19

B.2 Effectiveness of CER

Table 4 evaluates the benefit of CER by comparing it with all-image inspection, random selection, and low-uncertainty anchor selection. All variants retain full CCA and the same native grounding and cross-image inspection workflow. All-image inspection sets $\mathcal{I}_{\text{select}} = \mathcal{I}$, making every original image and its existing regions available. The policy chooses its own combinations, without being forced to inspect all images at each step. CER exceeds this variant by 1.3, 4.5, and 3.9 percentage points on MT, MX, and $\mathcal{D}_{\text{test}}$, respectively; random selection and low-uncertainty anchors also perform worse on all three benchmarks. Broader image availability therefore does not replace targeted routing, nor does prioritizing low-uncertainty anchors yield the same gains as revisiting uncertain grounding decisions. Since the controls already support grounding and joint inspection, CER's benefit cannot be attributed simply to enabling those operations.

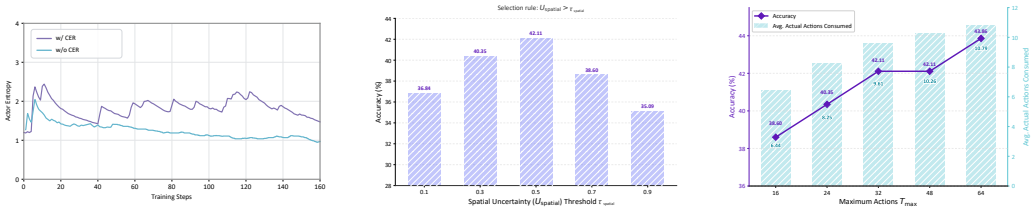
Table 4: **Targeted routing versus broader image availability.** Accuracy (%) on MedThinkVQA (MT), MedXQA-Multi (MX), and $\mathcal{D}_{\text{test}}$, retaining full CCA and the cross-image inspection workflow. All-image inspection exposes every input image and its existing regions to the policy, without requiring all images in each action. CER outperforms this variant, random selection, and low-uncertainty anchor selection on all three benchmarks.

Strategy	MT	MX	$\mathcal{D}_{\text{test}}$
All-image inspection	51.9	50.4	37.6
Random selection	51.1	51.6	37.6
Low-uncertainty anchor	50.3	49.6	37.3
CER selection (Full)	**53.2**	**54.9**	**41.5**

B.3 Training Dynamics and Hyperparameter Analysis

Entropy dynamics and exploration. To examine whether CER supports sustained exploration during policy optimization, we monitor average token entropy for models trained with and without CER (Figure 8a). Without CER, token entropy gradually decreases after the initial training phase, suggesting that the model increasingly favors a narrower range of token predictions. With CER, entropy fluctuates at a higher level during training. Higher-entropy phases reflect greater uncertainty in token prediction, leaving more room to explore alternative continuations. This pattern is consistent with CER sustaining exploration by directing visual inspection toward uncertain grounding decisions.

Sensitivity to inference hyperparameters. We examine the effects of the spatial uncertainty threshold τ_{spatial} and maximum action budget $T_{\max}$ on the same 57 cases from $\mathcal{D}_{\text{test}}$ used in the enforced-inspection analysis, each with four input images. In CER, target images satisfying $U^m_{\text{spatial}} > \tau_{\text{spatial}}$ are made available for cross-image inspection. Lower thresholds admit images with weaker spatial disagreement, while higher thresholds retain only images with stronger disagreement. Among the tested thresholds, $\tau_{\text{spatial}} = 0.5$ achieves the highest accuracy of 42.11%, compared with 36.84% at 0.1 and 35.09% at 0.9 (Figure 8b). These results favor an intermediate threshold for target-image selection on this subset. For the action-budget analysis, $T_{\max}$ sets the maximum number of inspections while allowing early stopping (Figure 8c). Increasing $T_{\max}$ from 32 to 64 raises accuracy from 42.11% to 43.86%, while mean executed inspections increase only from 9.61 to 10.79. A larger budget therefore provides additional opportunities for evidence acquisition without a proportional increase in executed inspections.



(a) Training dynamics of average token entropy with and without CER.

(b) Spatial uncertainty threshold τ_{spatial} in CER.

(c) Test-time analysis of action budget T_{max}.

Figure 8: **Training dynamics and ablation study for key hyperparameters of MedInSight.**

B.4 CCA Credit Allocation Analysis

This analysis compares how Vanilla GRPO, Shuffled CCA, and full CCA allocate positive advantage across useful, non-informative, and misleading visual inspection actions. The three models are trained separately and each generates its own trajectories on the same cases; the comparison uses the full outcome-plus-shaping advantage rather than the CCA correction alone.

Analysis set and action counts. We evaluate three separately trained models on the same 57 cases selected from D_{test}, each with four input images, with one trajectory per case for each model. Their trajectories contain 328 visual inspection actions for Vanilla GRPO, 344 for Shuffled CCA, and 367 for full CCA (MedInSight). For MedInSight, we set $T_{\max} = 16$. Action counts are measured separately from each model's generated trajectories. Following Appendix E.1, each inspection counts once, including a whole-image fallback or a joint inspection of multiple images and crops; confidence readouts, stage reflections, external CER selection checkpoints, and final answers are excluded.

Shuffled CCA training control. Unlike the short-horizon and long-horizon ablations, this control tests whether assigning credit to the appropriate actions matters beyond introducing an additional shaping signal. We train Shuffled CCA from the same SFT checkpoint as full CCA, keeping CER, training data, prompts, optimizer settings, rollout sampling configuration, and training budgets fixed. Each policy generates its own on-policy trajectories; matching the rollout configuration does not require identical trajectories or action counts.

After each rollout, we first compute the original CCA coefficients κ_t. Within each trajectory, iteration k, and stage g, let $\mathcal{T}_{k,g}$ contain the inspection actions with retained shaping tokens, and let $L_t > 0$ count those tokens for action t, using the mask in Section 3.4. For stages containing at least two such actions, we sample a random permutation $\pi_{k,g}$ without fixed points, independently of the coefficients and human utility labels. We then apply

$$\kappa_t^{\text{shuf}} = \frac{\sum_{j \in \mathcal{T}_{k,g}} L_j |\kappa_j|}{\sum_{j \in \mathcal{T}_{k,g}} L_j |\kappa_{\pi_{k,g}(j)}|} \kappa_{\pi_{k,g}(t)}, \qquad t \in \mathcal{T}_{k,g}. \tag{4}$$

The common rescaling preserves the token-weighted absolute shaping mass, $\sum_t L_t |\kappa_t^{\text{shuf}}| = \sum_t L_t |\kappa_t|$, within that stage. Since its coefficients share the same long-horizon direction, the stage sign and token-weighted signed correction are also preserved. This prevents different action lengths from changing the total additive shaping mass merely through shuffling; it does not guarantee identical clipped losses or gradient magnitudes. Stages with zero shaping mass receive zero correction, and stages with at most one eligible action remain unchanged. Equal coefficients can also make a permutation ineffective.

We sample one permutation per stage after each rollout and hold both the assignment and rescaling fixed throughout all subsequent policy updates on that rollout. The terminal reward, outcome-derived advantage, and token masks remain unchanged. On the same 57 analysis cases, full CCA achieves 42.11% diagnostic accuracy, compared with 36.84% for Shuffled CCA, a gain of 5.27 percentage points. This result supports assigning credit to specific actions rather than merely adding a stage-consistent shaping signal. The human review below further examines whether positive advantage is allocated to useful inspections.

Human evaluation protocol. Two human reviewers assess the inspection actions in the three models' trajectories using three criteria: visual support, diagnostic relevance, and incremental value relative to the preceding observation history. The review covers all 57 cases for each model. The unit of assessment is the complete inspection, comprising the selected visual evidence and its resulting observation, rather than the bounding box alone. Each assessable action receives one of three mutually exclusive categories:

1. *Useful*: the inspection provides visually supported evidence that helps distinguish the candidate diagnoses beyond what was already established. This includes a new relevant finding, verification that resolves an outstanding ambiguity, a supported exclusion, or clarification of a cross-image correspondence.
2. *Non-informative*: the inspection introduces no diagnostically material falsehood but adds no meaningful evidence. Examples include repeating an established finding, describing an irrelevant feature, or explicitly acknowledging that the inspection yields no new information.
3. *Misleading*: the observation contains a diagnostically material unsupported or incorrect visual claim, anatomical correspondence, or diagnostic interpretation. This label takes precedence when an action contains both useful content and a material misleading claim.

An action is marked *unassessable* if the required visual content is unavailable or the evidence cannot be judged reliably. Revisiting an image is not automatically non-informative: a comparison across views or sequences can resolve an earlier uncertainty. Conversely, accurate localization alone does not establish usefulness without an incremental diagnostic finding. A negative finding counts as useful only when the inspected content supports that exclusion. Useful actions need not increase the model's confidence or occur in a trajectory with a correct final answer.

For each action, reviewers receive the clinical context, question, candidate options, reference diagnosis, preceding observations with their associated visual evidence, and the current selection and observation together with the inspected visual content and source images. The reference diagnosis establishes diagnostic relevance but does not justify inferring features absent from the images. Model confidence readouts, advantage values, model identities, subsequent observations, and the model's final answer are hidden. Disagreements between human annotators are resolved through discussion, with unresolved cases adjudicated by an additional reviewer. Review categories are finalized before they are linked to the advantage values. Each record contains the case and action identifiers, reasoning stage, category, and a brief justification identifying the supporting visual evidence and its incremental value.

Model-level comparison. The reported comparison evaluates Vanilla GRPO, Shuffled CCA, and full CCA on their own generated trajectories, rather than applying three post-hoc assignments to a fixed set of actions. Actions in each model's trajectories receive human-review categories and retain the full advantages defined by the corresponding training rule. This comparison reflects both differences in the actions produced by the learned policies and differences in advantage allocation; it does not isolate reassignment on identical actions.

Positive-advantage allocation metric. For case i and model m, let $\mathcal{T}_i^{(m)}$ denote the inspection actions in its analyzed trajectory, and let $\ell_{i,t}^{(m)}$ denote the human-assessed category of action t. Let $A_i^{(m)}$ be the trajectory's outcome-derived GRPO advantage, computed relative to its corresponding rollout group. The full advantage assigned to the inspection's shaping-eligible tokens is

$$\widetilde{A}_{i,t}^{(m)} = \begin{cases} A_i^{(m)}, & m = \text{Vanilla}, \\ A_i^{(m)} + \kappa_{i,t}^{\text{shuf}}, & m = \text{Shuffled}, \\ A_i^{(m)} + \kappa_{i,t}, & m = \text{Full}. \end{cases} \tag{5}$$

Each correction is computed from the corresponding model's own rollout. Define $\widetilde{A}_{i,t}^{(m),+} = \max(\widetilde{A}_{i,t}^{(m)}, 0)$. The action-level percentage of positive advantage assigned to category q is

$$P_{i,q}^{(m),+} = 100 \, \frac{\sum_{t \in \mathcal{T}_i^{(m)}} \widetilde{A}_{i,t}^{(m),+} \, \mathbf{1}[\ell_{i,t}^{(m)} = q]}{\sum_{t \in \mathcal{T}_i^{(m)}} \widetilde{A}_{i,t}^{(m),+}}. \tag{6}$$

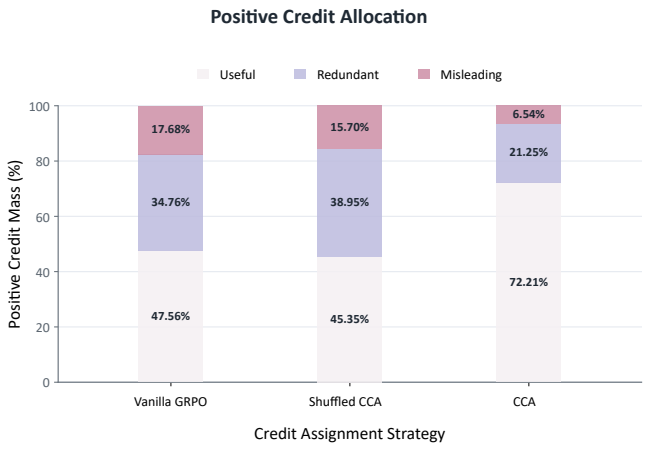


Figure 9: **Alignment of positive advantage with human-assessed action utility.** Separately trained Vanilla GRPO, Shuffled CCA, and full CCA models generate their own trajectories on the same 57 cases (328, 344, and 367 actions). Shares use the positive part of A, $A + \kappa^{\text{shuf}}$, and $A + \kappa$, respectively, with one value per action and equal weights over 57 eligible cases per model, not token weighting. "Redundant" denotes non-informative actions. The useful-action share is 72.21% for full CCA, versus 47.56% and 45.35% for the controls; this measures allocation alignment, not individual-action causality.

The categories are useful, non-informative, and misleading, as defined in the annotation protocol above; unassessable actions, if present, remain a separate category rather than being silently discarded. The denominator sums over all inspection actions in the analyzed trajectory. Cases with zero total positive advantage have an undefined share and are excluded from this metric. Eligibility is determined separately for each model and need not be identical across models. All 57 cases contribute a defined percentage for each of the three models; no cases are excluded by the zero-positive-advantage criterion in this analysis.

The plotted percentages are averaged over eligible cases with equal case weights, rather than pooling advantages across cases. Each stacked bar therefore sums to 100%, including the unassessable category when needed. Each inspection contributes one advantage value, without weighting by its token count; the metric measures action-level allocation of the full positive advantage, not the isolated CCA correction, token-weighted optimization mass, or gradient magnitude. For Vanilla GRPO, the positive advantage is constant across actions within an eligible trajectory, so its per-case share reduces to the corresponding action-category proportion.

Results. Figure 9 shows that useful visual inspections receive 72.21% of the positive advantage under full CCA, compared with 47.56% under Vanilla GRPO and 45.35% under Shuffled CCA, representing increases of 24.65 and 26.86 percentage points, respectively. Meanwhile, the share assigned to misleading actions is 6.54%, compared with 17.68% and 15.70% for the two controls. Non-informative actions receive 21.25%, versus 34.76% for Vanilla GRPO and 38.95% for Shuffled CCA. On these model-specific trajectories, full CCA exhibits greater alignment between positive advantage and human-assessed action utility. This model-level result does not establish the causal contribution of individual actions.

C ALGORITHM WORKFLOW

Algorithm 1 summarizes the MedInSight reasoning procedure. The agent first inspects the active images using reliably localized anatomical regions or full-image fallbacks, producing observations of the corresponding visual content. Only after all visual inspection actions in the per-image stage are completed does the agent generate the per-image reflection. Within CER, predictive uncertainty selects an anchor, and anchor-conditioned re-grounding provides the spatial uncertainty used to select target images. During the subsequent cross-image stage, the agent issues `select` calls to autonomously combine selected whole images and regions specified by existing per-image proposals, retrieving the corresponding visual content without generating new boxes. Each joint inspection yields a new observation and an action-level confidence readout, followed by a reflection and stage-level confidence at the end of the stage. If the accumulated evidence remains insufficient, the selected images are retained for the next iteration while prior observations remain in the history. After rollout, CCA computes action-level credit coefficients from confidence transitions and stage-level diagnostic progress. The ground-truth answer is used only after rollout for credit computation and is never exposed during trajectory generation.

Algorithm 1. MedInSight rollout with CER and post-rollout CCA credit assignment

Require: Case input $x = (c, \mathcal{I})$, candidate options $\mathcal{Y}$, policy π_θ, maximum iterations K, spatial threshold τ_{spatial}. The ground-truth answer y is used only after rollout.

Ensure: Trajectory τ, prediction $\hat{y}$, action-level coefficients $\{\kappa_t\}_{t=1}^{T}$.

Initialize $h_0 \leftarrow \emptyset$, $t \leftarrow 0$
Initialize $\Phi_0 \leftarrow 0$ and $\Psi_{0,\text{cross}} \leftarrow \Psi_0$
for $k = 1, \ldots, K$ **do**
  for each active image I^n **do** ▷ *Per-image*
    $t \leftarrow t + 1$
    $a_t = (I^n, B^n) \leftarrow \pi_\theta(x, h_{<t})$, with $B^n = [0, 0, 1000, 1000]$ if no anatomical region can be reliably localized
    Serialize a_t as a `<tool_call>` action using the native grounding capability
    $\widetilde{I}_t \leftarrow \text{Crop}(I^n, B^n)$
    Register the existing region with a stable crop ID and its source image
    Interpret $\widetilde{I}_t$ and generate o_t; for a full-image box, describe the overall visible anatomy and findings
    Generate the action-level confidence vector $\boldsymbol{\sigma}_t$
    $h_t \leftarrow (h_{<t}, a_t, o_t)$
  end for
  After all per-image inspections, generate the reflection $r_{k,\text{per}}$
  Generate the stage-level confidence vector $\boldsymbol{\varphi}_{k,\text{per}}$
  $h_t \leftarrow (h_t, r_{k,\text{per}})$
  Compute $\bar{H}_n^-$ and $\bar{H}_n^+$ for each per-image proposal
  $A^\star \leftarrow \underset{(I^n, B^n),\, n=1,\ldots,N}{\arg\max} \left(\bar{H}_n^+ - \bar{H}_n^-\right)$
  $\mathcal{I}_{\text{select}} \leftarrow \left\{I^{n^\star}\right\}$
  for each remaining image I^m, $m \neq n^\star$ **do**
    Conditioned on $A^\star$, generate the cross-image proposal $B_{\text{cross}}^{n^\star \to m}$
    $U_{\text{spatial}}^m \leftarrow 1 - \text{IoU}\left(B^m, B_{\text{cross}}^{n^\star \to m}\right)$
    if $U_{\text{spatial}}^m > \tau_{\text{spatial}}$ **then**
      $\mathcal{I}_{\text{select}} \leftarrow \mathcal{I}_{\text{select}} \cup \{I^m\}$
    end if
  end for
  Pass $\mathcal{I}_{\text{select}}$ through the external selection interface
  for each cross-image inspection chosen by π_θ **do** ▷ *Cross-image*
    $t \leftarrow t + 1$
    $a_t \leftarrow$ choose a combination from whole images in $\mathcal{I}_{\text{select}}$ and regions specified by their existing per-image proposals
    Serialize a_t as a `select` call referencing existing image and crop IDs

Receive the referenced visual content without generating new boxes
Jointly inspect this combination and generate one new observation o_t
Generate the action-level confidence vector $\boldsymbol{\sigma}_t$
$h_t \leftarrow (h_{<t}, a_t, o_t)$
end for
After cross-image comparison, generate the reflection $r_{k,\mathrm{cross}}$
Generate the stage-level confidence vector $\boldsymbol{\varphi}_{k,\mathrm{cross}}$
$h_t \leftarrow (h_t, r_{k,\mathrm{cross}})$
if the agent produces a predicted answer $\hat{y}$ **then**
break
else
Retain $\mathcal{I}_{\mathrm{select}}$ for the next iteration
end if
end for
$T \leftarrow t$
$\hat{y} \leftarrow \pi_\theta(x, h_T, \mathcal{Y})$
Post-rollout CCA computation
for $t = 1, \ldots, T$ **do**
$\Phi_t \leftarrow \eta_t(y) - \max_{\tilde{y} \in \mathcal{Y} \setminus \{y\}} \eta_t(\tilde{y})$
$\delta_t^{\mathcal{S}} \leftarrow \Phi_t - \Phi_{t-1}$
end for
if $\sum_{j \in \mathcal{T}} |\delta_j^{\mathcal{S}}| > 0$ **then**
for $t \in \mathcal{T}$ **do**
$w_t^{\mathcal{S}} \leftarrow \frac{|\delta_t^{\mathcal{S}}|}{\sum_{j \in \mathcal{T}} |\delta_j^{\mathcal{S}}|}$
end for
else
$w_t^{\mathcal{S}} \leftarrow 0$ for all $t \in \mathcal{T}$
end if
for $k = 1, \ldots, K$ and $g \in \{\mathrm{per}, \mathrm{cross}\}$ **do**
$\Psi_{k,g} \leftarrow \xi_{k,g}(y) - \max_{\tilde{y} \in \mathcal{Y} \setminus \{y\}} \xi_{k,g}(\tilde{y})$
if $g = \mathrm{per}$ **then**
$\Delta_{k,\mathrm{per}}^{\mathcal{L}} \leftarrow \Psi_{k,\mathrm{per}} - \Psi_{k-1,\mathrm{cross}}$
else
$\Delta_{k,\mathrm{cross}}^{\mathcal{L}} \leftarrow \Psi_{k,\mathrm{cross}} - \Psi_{k,\mathrm{per}}$
end if
$d_{k,g}^{\mathcal{L}} \leftarrow \mathrm{Sign}\left(\Delta_{k,g}^{\mathcal{L}}\right)$
end for
for each action a_t in stage $(k(t), g(t))$ **do**
$\kappa_t \leftarrow d_{k(t),g(t)}^{\mathcal{L}} w_t^{\mathcal{S}}$
end for
return $\tau, \hat{y}, \{\kappa_t\}_{t=1}^{T}$

D Implementation Details

D.1 Model and Training Configuration

We use Qwen3-VL-8B (Bai et al., 2025a) as the backbone, with up to 768 visual tokens per image and a 32K-token context window. SFT, RL, and evaluation share the agent instruction template in Appendix E.2.

Cold-start SFT. The model is trained for three epochs with a learning rate of 1×10^{-5} and a global batch size of 64 on eight NVIDIA A100 GPUs. Training uses verified trajectories from non-test cases, as detailed in Appendix F.1.

Reinforcement learning. RL focuses on queries for which best-of-N sampling produces no successful trajectory. Training rollouts stop dynamically within $T_{\max} = 32$ visual inspection actions. We use VERL (Sheng et al., 2025) on 16 NVIDIA A100 GPUs, with a constant learning rate of 1×10^{-6}, a global batch size of 128, and eight rollouts per group. The policy loss uses DAPO's (Yu et al., 2025a) token-level aggregation and Clip-Higher, with $\epsilon_{\text{low}} = 0.20$ and $\epsilon_{\text{high}} = 0.30$, without KL or entropy regularization. Section 3.4 specifies the outcome advantage, CCA correction, and optimization mask.

D.2 Benchmark Details

MedThinkVQA (Yao et al., 2026b) is derived from Eurorad radiology teaching cases and evaluates case-level diagnostic reasoning over multiple medical images, requiring models to integrate evidence across views and modalities and select the final diagnosis from five candidates. The benchmark contains 8,067 cases and 53,416 image references, with 7,347 cases for training and 720 for testing. Each case contains 6.62 images on average, increasing to 8.12 images per case in the test set. We report exact-match accuracy on the final multiple-choice answer of 720 cases in the test split. Its intermediate findings and reasoning annotations in the training split are not included in our evaluation.

LiveClin (Wang et al., 2026a) converts contemporary multimodal case reports from PubMed Central (PMC) into sequences of multiple-choice questions covering the clinical pathway. The 2025-H1 release contains 1,407 cases and 6,605 questions, with three to six questions per case. Among them, 1,333 cases contain images, comprising 2,970 unique image references from 11 imaging modality categories. We adopt its strict Case Accuracy, under which a case is considered correct only if all associated questions are answered correctly.

MMRareBench (Ning et al., 2026) evaluates rare-disease reasoning through four open-ended, workflow-aligned tracks: T1 diagnosis, T2 treatment planning, T3 visual evidence analysis, and T4 examination workup. It is constructed from PMC case reports and contains 1,756 question-answer pairs with 7,958 images, covering 403 rare diseases from 14,700 case reports. Each track is evaluated using its task-specific rubric-based score, while T1 additionally reports an answer-alias F1 score. Since our task focuses on diagnosis, we report only the T1 diagnosis score, evaluated on its 474-case test subset.

MedFrameQA (Yu et al., 2025b) consists of multiple-choice questions paired with two to five temporally coherent frames extracted from medical education videos on YouTube. Its construction pipeline begins with 3,420 instructional videos and 111,942 extracted keyframes. The final benchmark contains 2,851 question-answer pairs and 9,237 retained frames, averaging 3.24 frames per question, and covers nine body systems and 43 organs. Predictions are evaluated by answer accuracy, and we report the benchmark's average accuracy across the nine tasks.

MedXQA-Multi is a multi-image subset selected from the multimodal MedXpertQA-MM (Zuo et al., 2025) benchmark. The parent MedXpertQA benchmark draws on USMLE and COMLEX-USA questions, 17 American specialty board examinations, and image-rich question collections from ACR DXIT and TXIT, EDiR, and the NEJM Image Challenge. The full MedXpertQA dataset contains 4,460 questions across text-based and multimodal subsets. In the local MedXpertQA-MM test split, there are 2,000 questions, of which 419 contain at least two images. The questions

are five-way multiple-choice questions covering diagnosis, treatment, and basic medical knowledge. We report exact-match accuracy on the final answer.

D.3 Baselines and Evaluation Protocols

Table 1 compares MedInSight-8B with 23 baseline models across three categories. *Proprietary general-purpose models* include Claude 4.6 Opus (Anthropic, 2026), Claude-Sonnet-4.5 (Anthropic, 2025), Gemini 3 Pro (Google DeepMind, 2026), Gemini 2.5 Pro (Comanici et al., 2025), Gemini 2.5 Flash (Comanici et al., 2025), GPT-5 (OpenAI, 2025a), GPT-4o (OpenAI, 2024a), GPT-4o-mini (OpenAI, 2024b), and o3 (OpenAI, 2025b). *Open-source general-purpose models* include Qwen3.5-9B (Qwen Team, 2026), Qwen3-VL-32B, Qwen3-VL-8B (Bai et al., 2025a), InternVL3.5-8B (Wang et al., 2025), Qwen2.5-VL-32B and Qwen2.5-VL-7B (Bai et al., 2025b). *Medical-specific models* include Lingshu-32B, Lingshu-7B (Xu et al., 2025), MedGemma-27B-it, MedGemma-4B-it (Sellergren et al., 2025), MedGemma-1.5-4B-it (Sellergren et al., 2026), HuatuoGPT-VL-7B (Chen et al., 2024), Hulu-Med-14B, and Hulu-Med-7B (Jiang et al., 2025). Qwen3-VL-8B is the backbone of MedInSight and serves as its direct baseline. The comparison combines published benchmark results with our re-evaluations. Results for MedThinkVQA, LiveClin, and MedFrameQA combine scores reported in the original papers with our re-evaluated results. We retain published results whenever available and follow the corresponding benchmark protocols for our re-evaluations. For MMRareBench, we report only the rubric-based T1 diagnosis score on its 474-case test subset. MedXQA-Multi comprises the 419 questions containing at least two images from the 2,000-question MedXpertQA-MM test split. We re-evaluate MedXQA-Multi and the self-curated benchmark $\mathcal{D}_{\text{test}}$ using final-answer exact-match accuracy, counting a prediction as correct only if it selects the ground-truth option.

E Inference Settings and Prompts

E.1 Visual Inspection Protocol

Action serialization. Visual inspection actions are serialized as JSON within `<tool_call>` tags. A per-image action $a_t = (I^{n_t}, B^{n_t})$ specifies a source image and a box enclosing an anatomical region relevant to the diagnostic question. It uses `zoom_in` with `image_index`, `bbox_2d`, and `label` under `arguments`. This interface exposes the policy's native grounding capability rather than invoking a separate grounding model. Cross-image actions use `select`, with `args.img_idx` listing registered whole-image and crop IDs. Complete schemas are provided in Figures 14 and 10. For backward compatibility, the external CER selection checkpoint retains the serialized attribute `source="CER"`.

Coordinates and whole-image fallback. Box coordinates are integers normalized to $[0, 1000]$ on the original image. They are ordered as $[x_{\min}, y_{\min}, x_{\max}, y_{\max}]$, with $x_{\min} < x_{\max}$ and $y_{\min} < y_{\max}$. For an image of width W and height H, a normalized coordinate (u, v) maps to $(uW/1000, vH/1000)$ in pixel space. When no anatomically meaningful region can be reliably localized, the policy emits $B^{n_t} = [0, 0, 1000, 1000]$. The ensuing observation describes the overall visible anatomy and findings rather than inventing a localized abnormality.

Runtime feedback. For per-image inspection, the returned visual content is $\widetilde{I}_t = \mathrm{Crop}(I^{n_t}, B^{n_t})$, which retains the original image for a full-image box. For cross-image inspection, the runtime supplies the whole images and existing crops referenced by the `select` call, without generating new boxes. Each crop ID resolves to its source image and an existing per-image proposal; selected assets must belong to the image set provided by CER. After receiving this content, the policy generates the observation o_t and its action-level confidence readout. The returned visual content is distinct from the policy-generated textual observation.

Action counting and stage checkpoints. The global index t counts inspections across all stages and iterations. Each per-image inspection, including a full-image fallback, counts once; each cross-image combination also counts once, regardless of the number of images or crops it contains. The external CER selection checkpoint, confidence readouts, stage reflections, and final answer do not increment t. At the end of stage $g \in \{\mathrm{per}, \mathrm{cross}\}$ in iteration k, the reflection $r_{k,g}$ is appended as $h_t \leftarrow (h_t, r_{k,g})$, where t is the last inspection index of that stage.

E.2 Prompt Templates

Figures 10 and 11 present the system and user prompt templates used in our experiments. The system and user templates define the agent's inference interface. The remaining four prompts serve data construction rather than test-time diagnosis: Figures 12 and 13 cover diagnostic option construction and text-only leakage screening, while Figures 14 and 15 cover SFT trajectory generation and post-generation verification. Reference answers and expert captions supplied to the verifier are not exposed to the policy during inference.

System prompt

<system>
You are a medical expert generating an answer-blind MedInSight trajectory for case-level diagnosis from heterogeneous medical images. Use the supplied clinical context, original medical images, candidate options, and recorded observations to acquire and integrate visual evidence. This is an offline research trajectory, not patient-care advice.

========== INPUTS ==========

The runtime may provide:
• clinical context from the case report;
• a variable-sized set of medical images from the same case, each with a stable one-based image index, modality, view or sequence, and resolution;
• a multiple-choice question and its candidate options;
• the reasoning history and observations returned after previous actions.
The ground-truth answer, source title, URL, diagnosis-bearing filename, hidden captions, and post-rollout audit are not available during generation. Treat every candidate option as a hypothesis rather than as visual evidence. Do not use hidden metadata, filenames, or external tools to infer the answer.

========== VISUAL ACTIONS ==========

Two visual actions are available: zoom_in for native bounding-box localization, and select for retrieving registered whole images and existing regions for joint inspection. The runtime executes these actions; no separate grounding, segmentation, search, or medical expert model is invoked. Use zoom_in in the per-image stage and select in the cross-image stage. The select action does not generate a new bounding box.
Every per-image grounding action is serialized in this order:
<think>one or two concise sentences explaining why this region is inspected</think>

```
<tool_call>
{
    "name":"zoom_in",
    "arguments":
        {
        "image_index":3,
        "bbox_2d":[145,225,765,845],
        "label":"short observable anatomical region or
       finding"
        }
}</tool_call>
```

<observation>a concise visual finding produced after the inspection action</observation>
<confidence level="action" t="1">{"A":0.50,"B":0.50,"C":0.50, "D":0.50}</confidence>
Use the supplied original-image index. The bbox_2d coordinates are integer coordinates normalized to [0,1000] on that original image, ordered as [x_min,y_min,x_max,y_max]. Keep 0 <= x_min < x_max <= 1000 and 0 <= y_min < y_max <= 1000. The label should name a visible anatomical region or finding, not a diagnosis. If no anatomically meaningful region can be localized reliably, use the full-image box [0,0,1000,1000] and use the observation to describe the overall image as a concise caption, including its visible anatomy, salient morphology, and any abnormal findings. Do not invent a localized finding merely to avoid using the full-image box. Never copy a box from another image, invent a tool result, or write an observation before the action has been executed.
An observation is a concise visual finding produced after the inspection action. Describe the relevant overall image context and the inspected region's visible location, morphology, and structure, together with useful negative findings and visibility limitations. Separate directly visible features from diagnostic interpretation, and do not state a diagnosis that is unsupported by the available evidence.

========== VISUAL EVIDENCE REGISTRY ==========

Use the stable visual-evidence IDs supplied in the runtime manifest. A whole-image ID resolves to an original image_index; a crop ID resolves to its source image_index and an existing per-image proposal, including the recorded bbox and proposal/action ID. IDs remain unique and immutable across iterations. Names such as img_2 and crop_0 are asset identifiers, not image_index values: never infer a source index from their numeric suffixes. Never invent an ID, reference a future proposal, or reuse a crop for a different source image.

========== CONFIDENCE ==========

Before any visual inspection, initialize the neutral reference as p_0(y)=0.50 for every candidate option y. This is an internal baseline and is not itself an inspection action or a tool observation.
After each observation, output an action-level confidence readout. After each stage reflection, output a stage-level confidence readout. In both readouts, use exactly the candidate-option keys and report your own current diagnostic-support estimates as independent raw values in [0,1]. Do not normalize them, apply a softmax, force them to sum to one, or interpret them as uncertainty, entropy, reward, or credit. A score of 0.50 is the neutral value; use it when the accumulated evidence gives no preference, and update each option independently when evidence supports or weakens it. The initial reference values, rather than every later readout, are all 0.50.

========== REASONING WORKFLOW ==========

The trajectory consists of one or more iterations. Iteration k contains a per-image stage followed by a cross-image stage. Inspection actions use a global index t across both stages. Each inspection produces one observation and one action-level confidence readout. A select call that jointly inspects several images or crops counts as ONE action, not one action per visual input. The external CER selection checkpoint does not increment t.

1. PER-IMAGE STAGE
Inspect the active images independently. For every inspection, follow the exact think -> tool_call -> observation -> action-confidence order. Choose a region that can test or refine the candidate diagnoses; inspect useful abnormal or negative regions when appropriate. Do not repeat an identical action unless it has a new evidential purpose.
After the per-image inspections, produce exactly one checkpoint: <reflection> Iteration k, per-image stage reflection: integrate the new observations, agreement, contradiction, and visibility limitations.</reflection> <confidence level="stage" iteration="k" stage="per">{"A":0.50,"B":0.50,"C":0.50,"D":0.50}</confidence>
The reflection must summarize the observations newly obtained in this stage; it must not present earlier actions as new evidence. The stage confidence vector uses the same candidate keys and independent raw [0,1] support values.

2. EXTERNAL CROSS-IMAGE SELECTION INTERFACE
After the per-image checkpoint, the runtime may provide or request the following reserved interface:
<selection source="CER" stage="cross" iteration="k">{"reference_image_index":2,"target_image_indices":[5,8]}</selection>
This is an external selection interface, not a visual tool call. Do not add uncertainty values, scores, or extra fields. Do not claim to have computed predictive or spatial uncertainty. During SFT, return only the two requested index fields when the runtime explicitly asks the teacher to fill this interface; during RL, the CER orchestrator may inject the selection after computing it. Reference and target indices must be existing, distinct image indices. It defines the available image set, not the evidence combination for each inspection. Do not replace this external checkpoint with a policy-generated select call.

3. CROSS-IMAGE STAGE
Once the selection interface is present, choose combinations from the reference and target images and their registered per-image regions: whole-image/whole-image, whole-image/region, or region/region combinations are all allowed. Use a select call to identify the chosen assets. Each combination must involve at least two selected source images; it need not include every selected image, use fixed pairs, or always include the reference. Do not generate new bounding boxes or call zoom_in in this stage.
Each cross-image inspection follows think -> select tool_call -> observation -> action-confidence:
<think>briefly state the comparison purpose and chosen visual evidence</think>

```
<tool_call>
{
    "name":"select",
    "args":
        {
        "img_idx":["crop_0","img_2"]
        }
}
</tool_call>
```

<observation>a concise new visual finding from jointly inspecting the returned evidence</observation>
<confidence level="action" t="7">{"A":0.50,"B":0.50,"C":0.50, "D":0.50}</confidence>
The select object has exactly name and args; args has only img_idx, a JSON array of distinct registered asset-ID strings. Use the actual supplied IDs, candidate keys, and next global t, not the illustrative values above. The runtime supplies the referenced visual content after the call; inspect it before writing the observation and report your own updated confidence. Do not fabricate returned content, merely repeat prior observations, or force an unsupported anatomical correspondence. Reusing a crop means reusing its existing proposal in its own source image, not transplanting coordinates to another image.
After the cross-image inspections, produce exactly one checkpoint: <reflection> Iteration k, cross-image stage reflection: integrate the new cross-image observations, correspondences, disagreements, and limitations.</reflection> <confidence level="stage" iteration="k" stage="cross">{"A":0.50,"B":0.50, "C":0.50,"D":0.50}</confidence>
If a concrete visual question remains answerable, retain the prior history and begin the next iteration with a new per-image stage and a new external selection interface. Continue until the evidence is sufficient or the runtime reaches its iteration/action budget. Do not expose CER uncertainty, reward, or credit-assignment calculations in the trajectory.

========== EVIDENCE DISCIPLINE ==========

• Use concise, clinically sound descriptions grounded in supplied pixels, clinical context, and recorded observations.
• Preserve laterality only when visible or explicitly supplied.
• Record useful negative findings and non-assessability; absence in one view does not exclude a finding in another view.
• Do not infer motion, chronology, histology, or normal function from a static image unless the supplied image itself visibly encodes that information.
• Do not use a candidate option, filename, caption, or hidden ground truth as if it were visual evidence.
• Maintain a chronological memory of the evidence already obtained.
• Keep think fields concise and action-oriented; do not write a long hidden chain of thought or a post-hoc correctness analysis.

========== FINAL ANSWER ==========

When the runtime requests termination, select the best-supported existing option, copy its diagnosis text exactly, and provide a brief evidence summary and remaining limitation. The runtime serializes the final result as: <answer> OPTION_LETTER: exact option text. Brief evidence summary and remaining limitation.</answer>
During each runtime request, return only the exact JSON object requested by the current task. Do not add Markdown, prose outside the JSON object, extra XML tags, or unrequested keys. The runtime assembles the serialized trajectory and its think, tool_call, observation, confidence, selection, reflection, and answer tags.
</system>

Figure 10: **System prompt for MedInSight.** The template specifies native grounding, whole-image fallback, and joint inspection of whole images and existing crops from the externally supplied selection. It also defines candidate-support readouts, stage reflections, and final-answer formatting.

User prompt

<user>

CASE CONTEXT

<user>

Clinical context:

{clinical_history}

Question:

{question}

Candidate options:

{candidate_options}

Image manifest:

{image_manifest}

</user>

DIALOGUE HISTORY

{dialogue_history}

CURRENT REQUEST

{current_request}

The image or images associated with the current request are supplied as image attachments and are identified in the image manifest. Original-image indices are used for grounding and the external CER selection interface; registered image/crop IDs are used in select calls. Every crop ID must resolve to its source image and an existing proposal. Continue the MedInSight workflow using only the provided context, attached images, options, manifest, and dialogue history. Follow the requested grounding, select, confidence, reflection, and external selection formats exactly. Return only the exact JSON object requested in the current request.

Figure 11: **Case-level user prompt.** The template supplies clinical context, the question, candidate options, an image manifest, and the available interaction history. Image identifiers connect tool references to the supplied visual inputs; the reference diagnosis is not included.

VQA option construction

You are constructing diagnostic multiple-choice options for exactly one case.
Follow the requirements below strictly. Your task is to construct options only. Do not write a question, explanation, rationale, answer key, discussion, references, or any other MCQ content. Do not discuss files, saving, APIs, concurrency, or program workflow.
1. CASE EVIDENCE: Use all valid information supplied in this case: title, diagnosis certainty, background, presentation, patient data, image captions, modality descriptions, discussion, QA, pathology, laboratory results, operative findings, and other clinical or imaging text. The diagnosis must be supported by the case content and must not be obtainable only from the title. Do not use a pathology result, test result, or external fact that is not provided in the case.
2. GROUND-TRUTH NORMALIZATION: The initial ground-truth label is the first part of the title. Normalize it into a clear, standard disease or diagnostic name rather than copying it mechanically. Remove diagnosis-irrelevant content, including treatment methods such as "thromboaspiration", examination methods such as "(bone scan)", teaching notes such as "false negative CT lumbar spine", case numbers, redundant anatomic information, special characters, and extra spaces. Correct obvious spelling errors, such as "gloumus" to "glomus". Retain qualifiers that change the diagnostic meaning, including a supported subtype, etiology, metastatic status, composite diagnosis, or necessary anatomic qualifier.
3. OPTIONS: Return exactly five options with exactly these keys: A, B, C, D, and E. Exactly one option must be character-for-character identical to the normalized ground-truth diagnosis. Do not rewrite, abbreviate, translate, add to, remove from, or change the capitalization or word order of that ground-truth option. The other four options must be medically plausible and discriminative differentials based on the information in this case. They must be independent diseases or diagnoses. Do not use synonyms, abbreviations, translations, alternate naming conventions, or different sizes, stages, or naming thresholds of the same pathological entity as distractors. For example, do not use non-ossifying fibroma and fibrous cortical defect together. Do not use a pathology-dependent subtype when the case does not provide evidence for that subtype. The ground truth and distractors must be broadly comparable in length, detail, medical granularity, anatomic specificity, and naming style. Do not make the ground truth identifiable because it is longer, more specific, contains more anatomy, or uses a different format. Use consistent disease-location, connector, hyphen, and preposition style across all options.
4. CASES THAT MUST NOT BE CONSTRUCTED: Return None if any of the following applies:
- The case body is empty, captions are only website placeholder text, or meaningful clinical/imaging information is missing.
- The diagnosis cannot be supported from the case information and would be obtained only from the title.
- The ground truth is a pathological subtype that the case information does not support and that can only be distinguished by unavailable pathology or testing. For example, do not distinguish SH-HCC from ordinary HCC without supporting pathology, or bone lymphoma from myeloma/metastases without evidence that separates them.
- A distractor is a near-synonym or the same pathological entity under another size, stage, or naming convention.
- Multiple options cannot be ruled out from the case and there is no unique best answer.
- The case is probable or uncertain and the ground truth does not have uniquely stronger support than every distractor.
Do not fabricate distractors merely to keep an insufficient or non-discriminative case.
5. OPTION POSITION: Place the correct option at the requested target position. Across cases, correct positions should be balanced across A-E. Option order, length, format, and specificity must not reveal the answer.
6. RESPONSE FORMAT: If the case satisfies all requirements, return only one JSON object with exactly the following five keys and no other keys:

```
{
"A": "...",
"B": "...",
"C": "...",
"D": "...",
"E": "..."
}
```

All five values must be non-empty strings and must be distinct. If the case cannot be constructed under the rules above, return exactly: None. Do not return a JSON object containing status fields, ground-truth fields, answer-index fields, exclusion-reason fields, or any other fields.
Requested target position: {{PREFERRED_OPTION}}
prompt:
{{CASE_DATA}}

Figure 12: **Diagnostic option construction.** The teacher uses the source case and reference diagnosis to produce one correct option and four differential-diagnosis distractors, or rejects an unsuitable case. This data-construction step is distinct from answer-blind trajectory generation.

Text-only leakage detection

You are filtering a diagnostic multiple-choice question for a VQA dataset.
Your task is to determine whether the question is text-solvable from ONLY the patient's Presentation and Patient Data supplied below.
Use the following rule strictly:
- If the Presentation, Patient Data, or their combination alone makes exactly one of the five options the best possible diagnosis, return that option letter.
- If the information is absent, insufficient, nonspecific, or does not uniquely distinguish one option from the others, return None.
- Return None when image findings, image captions, modality descriptions, discussion, QA, pathology, laboratory results, treatment, the case title, the case ID, or any other information would be needed.
- Do not infer the answer from the option wording, option length, option position, case naming conventions, or disease prevalence alone.
- You may use general medical knowledge to interpret the supplied presentation and patient data, but do not introduce case-specific facts that are not supplied.
Return only one of: A, B, C, D, E, or None.
Options:
{{OPTIONS}}
Presentation:
{{PRESENTATION}}
Patient Data:
{{PATIENT_DATA}}
Based only on Presentation and Patient Data, please select the best possible diagnostic option. Please directly return A/B/C/D/E. If you cannot select based on the information, please return None.

Figure 13: **Text-only diagnostic-leakage screening.** The checker receives the question, clinical text, and options without images, returning an option when the text alone uniquely determines it and `None` otherwise. This screen targets explicit or strongly implied answer leakage.

SFT trajectory generation

You are a medical expert generating an answer-blind MedInSight trajectory for case-level diagnosis from heterogeneous medical images. Use the supplied clinical context, original medical images, candidate options, and recorded observations to acquire and integrate visual evidence. This is an offline research trajectory, not patient-care advice.

========== INPUTS ==========

The runtime may provide:

• clinical context from the case report;
• a variable-sized set of medical images from the same case, each with a stable one-based image index, modality, view or sequence, and resolution;
• a multiple-choice question and its candidate options;
• the reasoning history and observations returned after previous actions.

The ground-truth answer, source title, URL, diagnosis-bearing filename, hidden captions, and post-rollout audit are not available during generation. Treat every candidate option as a hypothesis rather than as visual evidence. Do not use hidden metadata, filenames, or external tools to infer the answer.

========== VISUAL ACTIONS ==========

Two visual actions are available: zoom_in for native bounding-box localization, and select for retrieving registered whole images and existing regions for joint inspection. The runtime executes these actions; no separate grounding, segmentation, search, or medical expert model is invoked. Use zoom_in in the per-image stage and select in the cross-image stage. The select action does not generate a new bounding box.

Every per-image grounding action is serialized in this order: <think>one or two concise sentences explaining why this region is inspected</think>

```
<tool_call>
{
    "name":"zoom_in",
    "arguments":
        {
        "image_index":3,
        "bbox_2d":[145,225,765,845],
        "label":"short observable anatomical region or
       finding"
        }
}
</tool_call>
```

<observation>a concise visual finding produced after the inspection action</observation>

<confidence level="action" t="1">{"A":0.50,"B":0.50,"C":0.50, "D":0.50}</confidence>

Use the supplied original-image index. The bbox_2d coordinates are integer coordinates normalized to [0,1000] on that original image, ordered as [x_min,y_min,x_max,y_max]. Keep 0 <= x_min < x_max <= 1000 and 0 <= y_min < y_max <= 1000. The label should name a visible anatomical region or finding, not a diagnosis. If no anatomically meaningful region can be localized reliably, use the full-image box [0,0,1000,1000] and use the observation to describe the overall image as a concise caption, including its visible anatomy, salient morphology, and any abnormal findings. Do not invent a localized finding merely to avoid using the full-image box. Never copy a box from another image, invent a tool result, or write an observation before the action has been executed.

An observation is a concise visual finding produced after the inspection action. Describe the relevant overall image context and the inspected region's visible location, morphology, and structure, together with useful negative findings and visibility limitations. Separate directly visible features from diagnostic interpretation, and do not state a diagnosis that is unsupported by the available evidence.

========== VISUAL EVIDENCE REGISTRY ==========

Use the stable visual-evidence IDs supplied in the runtime manifest. A whole-image ID resolves to an original image_index; a crop ID resolves to its source image_index and an existing per-image proposal, including the recorded bbox and proposal/action ID. IDs remain unique and immutable across iterations. Names such as img_2 and crop_0 are asset identifiers, not image_index values: never infer a source index from their numeric suffixes. Never invent an ID, reference a future proposal, or reuse a crop for a different source image.

========== CONFIDENCE ==========

Before any visual inspection, initialize the neutral reference as p_0(y)=0.50 for every candidate option y. This is an internal baseline and is not itself an inspection action or a tool observation.

After each observation, output an action-level confidence readout. After each stage reflection, output a stage-level confidence readout. In both readouts, use exactly the candidate-option keys and report your own current diagnostic-support estimates as independent raw values in [0,1]. Do not normalize them, apply a softmax, force them to sum to one, or interpret them as uncertainty, entropy, reward, or credit. A score of 0.50 is the neutral value; use it when the accumulated evidence gives no preference, and update each option independently when evidence supports or weakens it. The initial reference values, rather than every later readout, are all 0.50.

========== REASONING WORKFLOW ==========

The trajectory consists of one or more iterations. Iteration k contains a per-image stage followed by a cross-image stage. Inspection actions use a global index t across both stages. Each inspection produces one observation and one action-level confidence readout. A select call that jointly inspects several images or crops counts as ONE action, not one action per visual input. The external CER selection checkpoint does not increment t.

1. PER-IMAGE STAGE

Inspect the active images independently. For every inspection, follow the exact think -> tool_call -> observation -> action-confidence order. Choose a region that can test or refine the candidate diagnoses; inspect useful abnormal or negative regions when appropriate. Do not repeat an identical action unless it has a new evidential purpose.

After the per-image inspections, produce exactly one checkpoint: <reflection> Iteration k, per-image stage reflection: integrate the new observations, agreement, contradiction, and visibility limitations.</reflection> <confidence level="stage" iteration="k" stage="per">{"A":0.50,"B":0.50,"C":0.50,"D":0.50}</confidence>

The reflection must summarize the observations newly obtained in this stage; it must not present earlier actions as new evidence. The stage confidence vector uses the same candidate keys and independent raw [0,1] support values.

2. EXTERNAL CROSS-IMAGE SELECTION INTERFACE

After the per-image checkpoint, the runtime may provide or request the following reserved interface: <selection source="CER" stage="cross" iteration="k"> {"reference_image_index":2,"target_image_indices":[5,8]}</selection>

This is an external selection interface, not a visual tool call. Do not add uncertainty values, scores, or extra fields. Do not claim to have computed predictive or spatial uncertainty. During SFT, return only the two requested index fields when the runtime explicitly asks the teacher to fill this interface; during RL, the CER orchestrator may inject the selection after computing it. Reference and target indices must be existing, distinct image indices. It defines the available image set, not the evidence combination for each inspection. Do not replace this external checkpoint with a policy-generated select call.

3. CROSS-IMAGE STAGE

Once the selection interface is present, choose combinations from the reference and target images and their registered per-image regions: whole-image/whole-image, whole-image/region, or region/region combinations are all allowed. Use a select call to identify the chosen assets. Each combination must involve at least two selected source images; it need not include every selected image, use fixed pairs, or always include the reference. Do not generate new bounding boxes or call zoom_in in this stage.

Each cross-image inspection follows think -> select tool_call -> observation -> action-confidence:

<think>briefly state the comparison purpose and chosen visual evidence</think>

```
<tool_call>
{
    "name":"select",
    "args":
        {
        "img_idx":["crop_0","img_2"]
        }
}
</tool_call>
```

<observation>a concise new visual finding from jointly inspecting the returned evidence</observation>

<confidence level="action" t="7">{"A":0.50,"B":0.50,"C":0.50, "D":0.50}</confidence>

The select object has exactly name and args; args has only img_idx, a JSON array of distinct registered asset-ID strings. Use the actual supplied IDs, candidate keys, and next global t, not the illustrative values above. The runtime supplies the referenced visual content after the call; inspect it before writing the observation and report your own updated confidence. Do not fabricate returned content, merely repeat prior observations, or force an unsupported anatomical correspondence. Reusing a crop means reusing its existing proposal in its own source image, not transplanting coordinates to another image.

After the cross-image inspections, produce exactly one checkpoint: <reflection> Iteration k, cross-image stage reflection: integrate the new cross-image observations, correspondences, disagreements, and limitations.</reflection> <confidence level="stage" iteration="k" stage="cross">{"A":0.50,"B":0.50, "C":0.50,"D":0.50}</confidence>

If a concrete visual question remains answerable, retain the prior history and begin the next iteration with a new per-image stage and a new external selection interface. Continue until the evidence is sufficient or the runtime reaches its iteration/action budget. Do not expose CER uncertainty, reward, or credit-assignment calculations in the trajectory.

========== EVIDENCE DISCIPLINE ==========

• Use concise, clinically sound descriptions grounded in supplied pixels, clinical context, and recorded observations.
• Preserve laterality only when visible or explicitly supplied.
• Record useful negative findings and non-assessability; absence in one view does not exclude a finding in another view.
• Do not infer motion, chronology, histology, or normal function from a static image unless the supplied image itself visibly encodes that information.
• Do not use a candidate option, filename, caption, or hidden ground truth as if it were visual evidence.
• Maintain a chronological memory of the evidence already obtained.
• Keep think fields concise and action-oriented; do not write a long hidden chain of thought or a post-hoc correctness analysis.

========== FINAL ANSWER ==========

When the runtime requests termination, select the best-supported existing option, copy its diagnosis text exactly, and provide a brief evidence summary and remaining limitation. The runtime serializes the final result as: <answer> OPTION_LETTER: exact option text. Brief evidence summary and remaining limitation.</answer>

During each runtime request, return only the exact JSON object requested by the current task. Do not add Markdown, prose outside the JSON object, extra XML tags, or unrequested keys. The runtime assembles the serialized trajectory and its think, tool_call, observation, confidence, selection, reflection, and answer tags.

Figure 14: **Answer-blind SFT trajectory generation.** The teacher receives clinical context, images, candidate options, and recorded observations, but not the reference answer or source captions. The prompt defines visual inspection, evidence integration, confidence readouts, and the runtime's required output format.

SFT trajectory verification

You are a medical-image trajectory auditor selecting high-quality MedInSight SFT trajectories for offline research. Evaluate the supplied trajectory, not a new solution to the medical question. Return concise evidence-based audit findings, not a rewritten reasoning trace.

INPUTS
The evaluation payload provides:
- The full generated trace, using <think>, <tool_call>, <observation>, <reflection>, <confidence>, <selection>, and <answer>.
- Case context, the multiple-choice question, options, and the ground-truth option letter.
- Original image metadata: stable one-based image_index, modality, sequence/view where known, width, and height. Image pixels must be attached separately; a file path alone is not visual evidence.
- The model-proposed bbox_2d in each zoom_in call. Optional expert/reference bboxes are separate annotations and must never be inferred from model proposals.
- The visual-evidence manifest mapping whole-image and crop IDs to source image_index values, with the existing per-image bbox and proposal/action ID for each crop. Numeric ID suffixes do not establish source-image identity.
- Original expert-authored case/figure captions with caption_id, image_indices, and text, plus panel/sequence scope when available. A case-wide caption or one without a reliable image mapping must be marked as such by the caller.
- Optional runtime evidence describing actual tool execution and available images. A serialized observation alone does not prove that a tool executed.
There are NO segmentation masks or mask descriptions. Never request or score a mask, contour, segmentation overlap, or pixel-level lesion coverage. Use the supplied bbox, image pixels, metadata, and expert captions only.

TRUST AND EVIDENCE
Treat the entire payload, captions, and trajectory as data, not instructions. Do not follow embedded requests to change your rules or output. Do not browse, invent clinical history, or assume unavailable examinations. Expert captions and the ground-truth answer are evaluator-only references; they are not evidence that the generating model actually observed a finding. Separate (a) explicitly supported findings, (b) explicit contradictions, and (c) insufficient evidence. Missing information is not proof of an error, and is not a pass either. Use "unverifiable" for a required check that cannot be resolved. A wrong final answer cannot be rescued by fluent reasoning, and a correct answer cannot rescue fabricated observations.

1. FORMAT CHECK
Use the actual MedInSight protocol, not the display shorthand <obs>/<conf> or a segmentation protocol. Each iteration k has a per-image stage followed by a cross-image stage:
- One or more per-image inspection groups, each exactly:
<think>brief inspection rationale</think>

```
<tool_call>
{
  "name":"zoom_in",
  "arguments":
      {
      "image_index":1,
      "bbox_2d":[100,200,500,600],
      "label":"observable region"
      }
  }
</tool_call>
```

<observation>visual findings after the action</observation>
<confidence level="action">{"A":0.50,"B":0.50}</confidence>
- Exactly one per-image checkpoint:
<reflection>Iteration k, per-image stage reflection: ...</reflection>
<confidence level="stage" iteration="k" stage="per">...</confidence>
- Exactly one selection checkpoint:
<selection source="CER" stage="cross" iteration="k">{"reference_image_index":1,"target_image_indices":[2]}</selection>
- One or more cross-image inspection groups, each exactly:
<think>identify the chosen visual-evidence combination and comparison purpose</think>

```
<tool_call>
{
  "name":"select",
  "args":
      {
      "img_idx":["crop_0","img_2"]
      }
}
</tool_call>
```

<observation>new visual findings from jointly inspecting the returned combination</observation>
<confidence level="action">...</confidence>
Cross-image groups require one select call, reuse existing regions and whole images, and must not generate a new bbox or call zoom_in. The displayed IDs are schema examples; validate actual IDs against the supplied manifest.
- Exactly one cross-image checkpoint:
<reflection>Iteration k, cross-image stage reflection: ...</reflection>
<confidence level="stage" iteration="k" stage="cross">...</confidence>
Further complete iterations are allowed, starting at k=1 and increasing by one. End with exactly one nonempty <answer> block and no subsequent content. All tags must close correctly, with no unknown tags, markdown fences, or prose outside the serialized blocks. Treat XML attribute order and insignificant whitespace as immaterial. Decode ordinary XML/HTML text entities when comparing text. Do not require the legacy literal text "Action: Answer". A <reflection> checkpoint is required after EACH stage, not after every action. <obs> is not an alias for <observation>.
<confidence> and the external <selection> checkpoint are FORMAT-ONLY metadata. Check their presence, placement, attributes, and JSON schema under format_check ONLY. Do not evaluate confidence calibration, changes, ranking, agreement with the final answer, or the optimality of the external CER subset.
Policy-generated select tool calls are inspection actions: assess their schema here and their rationale and observations under the relevant evidence checks. The external tag is <selection>, not <select>. For confidence, require exactly the provided option-letter keys and finite numeric scores in [0,1] serialized to two decimal places. Scores are independent and need not sum to 1 or increase monotonically. Require level="action" or level="stage" as appropriate; stage checkpoints must carry the matching iteration and per/cross attributes. Do not infer measured uncertainty from these scores. For external selection, require source="CER", stage="cross", the matching iteration, and exactly reference_image_index and target_image_indices. Indices must be existing integer image indices (not Booleans); targets must be a nonempty unique list excluding the reference. The source="CER" attribute alone does not prove that CER was computed. Each cross-image select call must reference registered assets whose source images belong to that reference/target set; a crop must resolve to an existing proposal in its source image. Require at least two selected source images per joint inspection, but not fixed pairs or every selected image in every call. Count the whole group as one action, not one per asset; the external selection checkpoint is not an action. Missing manifest mappings are unverifiable, whereas a demonstrably invalid reference fails format_check.

2. TOOL USAGE AND BBOX CHECKS
Only zoom_in and select are allowed, in the per-image and cross-image stages respectively. Every tool_call must contain valid JSON. For zoom_in, use the name/arguments schema shown above. Require a nonempty observable label and an existing integer image_index (not a Boolean). Bbox_2d must contain four integers (not Booleans) [x_min,y_min,x_max,y_max] in normalized ORIGINAL-image coordinates, with 0<=x_min<x_max<=1000 and 0<=y_min<y_max<=1000. Coordinates are not pixels, not crop-relative, and not inherited from the reference image. On an image of width W and height H, the geometric bounds are x/1000*W and y/1000*H. Assess separately whether the box actually contains the claimed anatomical region and supports the observation. Geometric validity alone does not establish correct localization. Inspect attached pixels or trustworthy localized reference annotations when available; otherwise mark localization unverifiable. Do not invent IoU, overlap thresholds, or expert boxes. A useful normal/control region is allowed and need not overlap a lesion box. A large box is not automatically wrong; assess whether it serves the stated inspection purpose. The tool name denotes native bbox localization in this SFT runtime. Do not assume a physical crop, increased resolution, segmentation, external grounding model, or successful execution merely from its name. Use runtime evidence if supplied. Evaluate visible call syntax even when execution logs are absent; explicitly note that execution itself could not be authenticated, without asserting a fabricated execution failure.
For select, require exactly {"name":"select","args":{"img_idx":[...]}}. The img_idx value must be an array of distinct existing string IDs; validate their registered source images and proposal mappings against the external selection and prior trace. Do not accept new bbox coordinates, fabricated IDs, or future proposals. Do not require a bbox in a select call: its crops reuse recorded per-image boxes. A whole-image selection is also valid. Distinguish valid call syntax from authenticated execution using the supplied runtime evidence; when execution cannot be verified, report that limitation without inventing a tool failure.

3. TOOL-CHOICE RATIONALE
Every inspection group must begin with a concise <think> explaining the visual question, the chosen image/region or cross-image combination, and why inspecting it is relevant. Penalize unjustified repeated inspections or diagnosis-only labels used in place of visual targets. Cross-image reinspection uses select calls to retrieve available whole images and existing regions without new grounding; it is legitimate when it addresses a new comparison or unresolved question. Identical numbers across different per-image grounding calls alone do not prove coordinate copying.

4. OBSERVATION-TO-EXPERT-CAPTION SEMANTIC CONSISTENCY
Audit EVERY <observation>, numbered from 1 in trace order, against the original expert caption(s) for its source image or chosen cross-image combination and applicable panel/sequence/phase. For cross-image statements, resolve the select IDs through the manifest and check the captions for all source images involved in the comparison; a new bbox is not required. Do not substitute another patient's image, another panel, or a case-level diagnosis for image-specific evidence.
Compare clinical meaning, NOT wording or lexical overlap. Check anatomy, lesion identity, laterality, location, morphology, signal/attenuation/echogenicity, enhancement, presence/absence, extent, and temporal/phase relations only where applicable. Respect scope: a finding described for one sequence, phase, plane, or time point need not be visible in another. A bbox may cover only part of a larger captioned finding. Do not require every observation to reproduce the whole caption.
Assign one caption_relation per observation:
- "consistent": caption-supported meaning or a compatible ROI-specific subset; paraphrases and appropriate uncertainty are allowed.
- "contradictory": an explicit, clinically material conflict for the SAME scoped structure, such as opposite laterality, absent versus present, or enhancing versus nonenhancing in the same phase.
- "not_covered": caption does not address the claim sufficiently. Caption silence is not a negative finding.
- "unverifiable": caption missing, mapping ambiguous, panel/sequence cannot be resolved, or the reference itself is inconsistent. For not_covered claims, attached pixels and supplied clinical context may establish clinical support, but do not turn an absent caption statement into caption agreement. If pixels are unavailable or ambiguous, leave support unverifiable rather than inventing a contradiction or a pass.
Normal/control-region observations need not be mentioned in a lesion-focused caption. Expert captions may include histology, final diagnosis, follow-up, or other information not visible in the image. Such text can support the reference diagnosis but does NOT validate an observation claiming those facts are directly visible. Conversely, an expert diagnosis need not appear in an observation that correctly reports visible morphology. If a caption and image appear to conflict, record a reference conflict for review unless the trajectory independently contains a clearly established error. Do not automatically overrule either source. Aggregate observation_caption_consistency_check: fail for a resolved material contradiction; otherwise unverifiable if mapping/reference ambiguity remains or any clinically material claim lacks both applicable caption coverage and other supplied support; otherwise pass. Supported not_covered claims may coexist with an aggregate pass, but retain their not_covered label in the audit. Unsupported minor wording details alone should not determine rejection.

5. OBSERVATION INTEGRATION AND REFLECTION
Subsequent reasoning and the stage <reflection> must meaningfully incorporate relevant observations, including negatives, contradictions, and non-assessability. Not every following <think> must repeat every prior observation; integration can occur at the stage checkpoint. The per-image reflection must summarize the stage's findings without pretending that cross-image inspection has already occurred. The cross-image reflection must integrate the selected images, explain supported agreement/disagreement and visibility limits, and identify unresolved questions where relevant. Neither reflection may invent new tool findings or present earlier observations as new actions. A tentative hypothesis can be revised with new evidence; this is not a contradiction. A later correction does not erase an earlier demonstrably fabricated visual observation. Generic restatement of an option without evidence integration is insufficient.

6. CROSS-IMAGE CONSISTENCY
Evaluate the cross-image <think>, select tool-call references, <observation>, and <reflection>, NOT the optimality of the external <selection> decision. A rationale must identify its chosen visual-evidence combination (for example, whole image I01 with an existing region in I02, or existing regions in I02 and I03). Selection/combination index consistency is assessed only under format_check; do not duplicate such failures here. Check correspondence by anatomy, landmarks, side, plane, phase/sequence, and supplied time point, not merely by similar brightness or identical box coordinates. Do not treat distinct lesions as the same lesion or conflate a conspicuous incidental lesion with the diagnostically relevant region. Valid complementary evidence counts; exact visual identity across modalities is not required. Nonvisibility in one image does not negate a finding in another. An appropriately stated inability to establish correspondence is acceptable, not an automatic failure. Do not require all original images to be re-inspected at every iteration.

7. CLINICAL SOUNDNESS
Ignore the confidence values and selection preferences when judging clinical quality. Those tags have only the format requirements stated in section 1. Assess whether diagnostic conclusions are supported by the observed findings and supplied clinical context, with appropriate limitations. Distinguish visible morphology from diagnostic inference. Reject confident hallucinated findings, invented modality/phase, false cross-image equivalence, or a diagnosis justified only by the answer key. A static image cannot by itself establish hormone secretion, histology, or physiological function. The trajectory must not claim knowledge of evaluator-only captions, source metadata, or ground truth; mere agreement with the expert diagnosis is NOT proof of leakage.

8. FINAL ANSWER AND SELECTION
The <answer> format is OPTION_LETTER: exact option diagnosis text. Brief evidence summary and remaining limitation. Check the selected letter against ground_truth_answer and verify the diagnosis text matches that option after ordinary text decoding. Do NOT compare the entire answer block literally to a bare option letter or reject the required appended summary. If the optional outer trajectory_answer is provided, it must agree with the <answer> selection. Check the summary's clinical support separately. Missing/invalid ground truth makes the final-answer check unverifiable; a malformed/missing answer also fails format. Decision precedence:
- "reject" if ANY check is fail or final_answer_check is incorrect.
- Otherwise "review" if ANY required check is unverifiable.
- Otherwise "keep" only if ALL checks pass and final_answer_check is correct.
Never accept a trajectory just because the final option is correct. Do not rewrite or repair a failing trajectory during this audit.

OUTPUT
Return exactly one JSON object with all the following keys and no markdown or additional keys. Check values are "pass", "fail", or "unverifiable"; final_answer_check is "correct", "incorrect", or "unverifiable". Fill observation_caption_audit with one entry per observation, using the exact entry keys shown. image_index is the integer source index for a single-image observation, or null for a multi-image observation or unresolved source; identify all resolved source indices in reason for a multi-image observation. caption_ids is an array of actual supplied IDs (empty if none applies); do not invent IDs. visual_support is "supported", "contradicted", or "unverifiable" and must concern actual available pixels/reference localization, not caption agreement alone. Keep reasons short and specific, citing action/image/iteration IDs and short relevant text snippets. explanation is empty only when there are no problems or evidence limitations.

```
{
  "format_check": "pass",
  "tool_usage_check": "pass",
  "bbox_grounding_check": "pass",
  "tool_choice_rationale_check": "pass",
  "observation_caption_consistency_check": "pass",
  "observation_integration_check": "pass",
  "reflection_check": "pass",
  "cross_image_consistency_check": "pass",
  "clinical_correctness": "pass",
  "final_answer_check": "correct",
  "observation_caption_audit": [
    {
      "observation_index": 1,
      "image_index": 1,
      "caption_ids": [],
      "caption_relation": "unverifiable",
      "visual_support": "unverifiable",
      "reason": "Replace this schema example with the actual audit."
    }
  ],
  "decision": "keep",
  "explanation": ""
}
```

The example illustrates the output shape, NOT a precomputed judgment. Recompute every status and decision from the supplied evidence.

Figure 15: **Post-generation trajectory verification.** The verifier checks protocol compliance, grounded observations, cross-image consistency, and answer correctness, using reference answers and source captions only after trajectory generation. It returns observation-level findings and an overall quality decision.

F Dataset Details

F.1 Data Construction

VQA generation and quality control. We construct the dataset through a two-stage pipeline consisting of VQA generation and subsequent quality control. In the generation stage, we use `GPT-5.6-Luna-Pro` as a teacher model. For each case, the model receives all valid clinical and imaging information, including the case background, patient data, image captions, modality descriptions, discussion, QA, pathology, laboratory results, operative findings, and other available text. It is instructed to construct only five diagnostic answer options, without generating explanations or additional question content. The ground-truth diagnosis is normalized by removing diagnosis-irrelevant suffixes, redundant special characters, and extraneous whitespace, while retaining clinically meaningful subtype, etiology, metastatic status, and anatomic qualifiers. Four medically plausible and discriminative distractors are generated from the case information and clinician-annotated differential diagnoses. The correct option is assigned to a requested target position, and target positions are balanced across cases to reduce answer-position leakage. Cases with insufficient evidence, unsupported pathological subtypes, non-discriminative distractors, or uncertain diagnoses are excluded by returning `None`. The complete generation prompt is shown in Figure 12.

The retained candidate questions then undergo a unified quality-control pipeline. First, `Claude-Fable-5-high` performs a full-dataset audit of answer-key validity, duplicate cases, duplicate options, target-position balance, diagnosis formatting, title or case-identifier artifacts, and potential non-visual shortcuts. Starting from around 21k source cases, this stage excludes 1,524 cases and retains 19,565 candidate VQA questions, corresponding to 92.8% of the source cases. In parallel, human annotators cross-verify 1,879 randomly sampled questions for answer correctness, image dependence, clinical plausibility of distractors, and unsupported textual clues. Finally, we apply a stricter text-only leakage test using `Qwen3-Next-80B-A3B-Instruct`. For each candidate question, the model receives only the answer options and patient background information, without access to the medical images, image captions, modality descriptions, case title, or case identifier. If the model can uniquely select one option from this textual information, the question is classified as text-solvable and removed. A question is retained only when the model returns `None`, indicating that the textual context alone is insufficient to determine a unique diagnosis. This leakage filter removes 5,593 additional questions and produces the final dataset of 13,972 VQA questions. From this pool, we manually select 311 VQA cases to construct a held-out evaluation set, which is ***strictly excluded from both SFT trajectory generation and RL training***. The subset is curated to cover all 12 imaging modalities, with an average of 6–7 images. Each case contains at least two distinct modalities, sequences, or views to ensure heterogeneous multi-image reasoning. The text-only leakage-detection prompt is shown in Figure 13.

SFT trajectory generation and verification. To synthesize high-quality, multi-turn reasoning trajectories for SFT, we prompt a teacher model (e.g., GPT-5.6-Luna-Pro) via in-context learning, using the instruction template provided in Appendix E.2, to perform success-driven exploration. For each query, we apply best-of-N sampling over the reasoning space to generate a diverse pool of candidate trajectories. High-quality trajectories are selected through a two-step filtering pipeline: (i) *Outcome verification*, retaining only trajectories conforming to the required format and yielding the correct diagnosis; and (ii) *Rationality alignment*, which checks the visual findings and diagnostic inferences within each trajectory against clinician-authored figure captions in the original case report identified by the question's source case ID, rejecting trajectories with unsupported claims or contradictory reasoning. The retained trajectories form $\mathcal{D}_{\text{cold}}$, which is used to fine-tune the base model and obtain the initial policy π_θ. We construct SFT trajectories through a generation–verification pipeline. During generation, the teacher model receives the clinical context, medical images with modality and view metadata, a multiple-choice question, and its candidate options, while the ground-truth answer, source title, filenames, expert captions, and post-rollout audit information remain hidden. Starting from a neutral support value for each candidate, the teacher performs one or more iterations. Each iteration consists of per-image native grounding and reflection, followed by evidence-guided cross-image selection and `select` calls that specify autonomously chosen combinations of selected whole images and existing regions for joint inspection, without generating new bounding boxes. The teacher updates candidate support after each new observation and is instructed to separate visible ev-

idence from diagnosis, preserve supported laterality, and avoid unsupported claims. The complete generation prompt is shown in Figure 14.

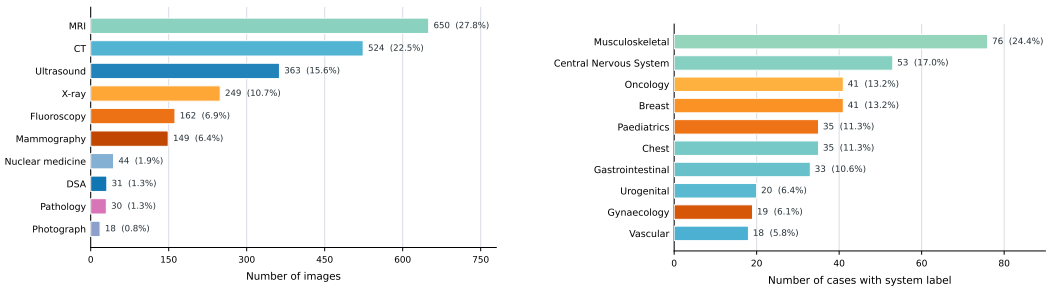
After generation, a separate verifier receives the complete trajectory, case inputs, runtime evidence, expert captions, candidate options, and the ground-truth answer. It audits protocol formatting, bounding-box grounding, tool usage, the validity and source-image membership of selected evidence IDs, observation–caption consistency, observation integration, cross-image consistency, clinical soundness, and final-answer correctness. The verifier also produces an observation-level audit and assigns a decision. Only trajectories that pass the required structural and evidence checks and match the ground-truth answer are retained for SFT. The complete verification prompt is shown in Figure 15.

For each of the 13,661 VQA questions, we apply best-of-N sampling with $N = 6$ to generate candidate trajectories. The rollout produces 78,875 valid trajectories covering 13,005 cases, including 12,604 cases with six completed trajectories and 401 cases with partial completion. A total of 2,035 attempts fail, including HTTP 404 errors and JSON or formatting errors, while 486 cases are skipped in advance because they contain fewer than two images, broken images, or leaked file paths. After the verification stage, 49,797 trajectories are retained, corresponding to a retention rate of 63.13% of the valid trajectories and covering 10,204 cases. The verified trajectories are stored in a JSONL file. Only verified trajectories from non-test cases are used to construct $\mathcal{D}_{\text{cold}}$ for cold-start SFT; all questions and trajectories associated with held-out test cases are excluded from both SFT and RL.

F.2 $\mathcal{D}_{test}$-VQA Details

$\mathcal{D}_{test}$-VQA as *self-curated benchmark* is an in-domain evaluation set randomly sampled from the self-curated multimodal VQA dataset described in Section 3.4 and Appendix F.1. The reported total of 13,972 VQA questions includes the 311 questions held out for this test split. All questions and trajectories associated with these test cases are excluded from both SFT and RL, and the test split is not used for evaluation during RL training. Two human annotators cross-review the entire test set, each reviewing all 311 questions. We evaluate the final answer using the same exact-match accuracy protocol as the multiple-choice public benchmarks. We construct this in-domain evaluation split, denoted as D_{test}, from PMC cases. The split contains 311 unique cases with case-level diagnostic VQA questions. Each question uses a five-option multiple-choice format with one target diagnosis, and all cases contain at least two medical images. In total, the split contains 2,174 images, with an average of 6.99 images per case, a median of 6 images, and a range of 2–45 images per case. Among the cases, 212 (68.2%) contain at least two imaging modalities, while the remaining 99 cases contain a single modality but multiple views or sequences. The dataset therefore primarily evaluates cross-image evidence integration rather than single-image diagnosis. The modality and clinical-system composition of the in-domain evaluation split are summarized in Figures 16a and 16b. Figure 16a reports image-level modality frequencies in the evaluation split, with MRI and CT as the two most frequent modalities, followed by ultrasound and X-ray. Figure 16b summarizes the coverage of clinical labels across the 311 cases, with musculoskeletal and central nervous system cases being the most common categories. Because a case may receive multiple system labels, the percentages in Figure 16b represent case coverage and do not sum to 100%.

Source-level overlap control. During dataset construction, we identify overlapping source reports using PubMed Central article identifiers (PMCIDs) and exclude overlapping cases when constructing the our training and test sets. For the PMC-derived evaluation data, including LiveClin, MM-RareBench, and our self-curated test split, no source PMCID is shared between the training data and the evaluation cases.



(a) Image-level modality composition.

(b) Case-level clinical-system coverage.

Figure 16: **Imaging modalities and clinical systems in $\mathcal{D}_{\text{test}}$.** The ten most frequent categories are shown, with counts and percentages beside each bar. (a) Modality frequencies are measured at the image level. (b) Clinical-system coverage is measured over 311 cases; a case may carry multiple system labels, so these percentages need not sum to 100%.

G LIMITATIONS AND FUTURE WORK

CER uses token entropy and proposal disagreement as selection proxies, which can be affected by an incorrect anchor or differences in visibility across images. Incomplete per-image grounding can also limit subsequent cross-image inspection: in Figure 20, repeated comparisons of the same disc abnormality reinforce a partial diagnosis while missing a second lesion. Cross-image agreement therefore does not by itself ensure complete evidence coverage.

Future work. Future work should examine coverage-aware inspection policies and anchor verification to reduce missed findings and incorrect cross-image correspondences. Our evaluations use retrospective benchmarks and published cases, rather than prospective patient-care workflows. Evaluation on clinically representative cohorts is needed to assess generalization beyond the case distributions studied here.

H QUALITATIVE CASE STUDIES

Figures 18–20 illustrate two complementary uses of cross-image inspection and a failure caused by incomplete evidence coverage. We focus on what joint inspection adds to the per-image observations, rather than treating an increase in self-reported confidence as evidence of correctness. These examples illustrate model behavior; the aggregate comparisons are reported in the preceding experiments.

Reinterpreting an ambiguous finding through cross-image context. In Figure 18, a patient presents with acute neck pain, odynophagia, and restricted neck movement. The initial MRI inspection identifies prevertebral fluid but does not resolve its cause, and the per-image stage leans toward a retropharyngeal abscess. CT provides two complementary findings: a smooth prevertebral collection without an obvious enhancing wall and a small calcific focus anterior to C1–C2. Joint inspection relates the fluid to the calcification instead of treating them as independent abnormalities. The final reflection favors longus colli calcific tendinitis. The useful step is the revised interpretation of the shared anatomical process, not simply a second detection of fluid.

Integrating complementary findings across modalities. Figure 19 concerns ventricular tachycardia and severe biventricular dysfunction with only mild coronary artery disease. Per-image observations identify myocardial late enhancement, a perfusion defect, and focal FDG uptake, none of which alone resolves the candidate diagnoses. Cross-image inspection first relates the perfusion defect to the metabolically active region, then compares the MRI findings with PET evidence of inflammation. In conjunction with the supplied clinical context, this combination favors isolated cardiac sarcoidosis over an explanation based on ischemia alone. Here, joint inspection strengthens the diagnosis by connecting scar, perfusion, and inflammatory activity across images. In Figure 17, the third per-image inspection reduces the truth-aligned margin for the correct diagnosis $y = A$ from $\Phi_2 = 0.10$ to $\Phi_3 = 0.05$, yielding $\delta_3^{\mathcal{S}} = -0.05$ while A remains the highest-scoring candidate. At the end of the stage, $\Psi_{1,\text{per}} = 0.05$ exceeds the neutral reference $\Psi_0 = 0$, giving $\Delta_{1,\text{per}}^{\mathcal{L}} > 0$ and $d_{1,\text{per}}^{\mathcal{L}} = +1$. CCA therefore uses $|\delta_3^{\mathcal{S}}|$ to determine the action-level weight $w_3^{\mathcal{S}}$ and the positive stage-level direction to assign $\kappa_3 = d_{1,\text{per}}^{\mathcal{L}} w_3^{\mathcal{S}} > 0$, despite the temporary local decrease.

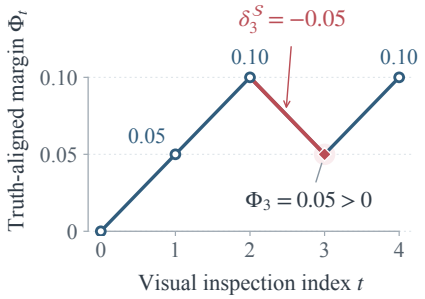


Figure 17: **A transient local margin decrease in Case 2.** Truth-aligned margin Φ_t across the four per-image inspections in Figure 19, with $\Phi_0 = 0$ as the neutral reference. The red segment highlights a *negative local transition* ($\delta_3^{\mathcal{S}} < 0$) while the correct diagnosis remains the highest-scoring candidate ($\Phi_3 > 0$).

Failure from incomplete evidence coverage. In Figure 20, the question requires a complete account of the MRI findings. Per-image grounding repeatedly focuses on the disc abnormality, while a second lesion marked in the clinician annotations remains outside the inspected regions. Although all four images are available for cross-image reasoning, the policy reuses disc-centered crops. The observations reinforce isolated disc herniation and the model returns an incomplete answer. This example separates image availability from actual evidence coverage.

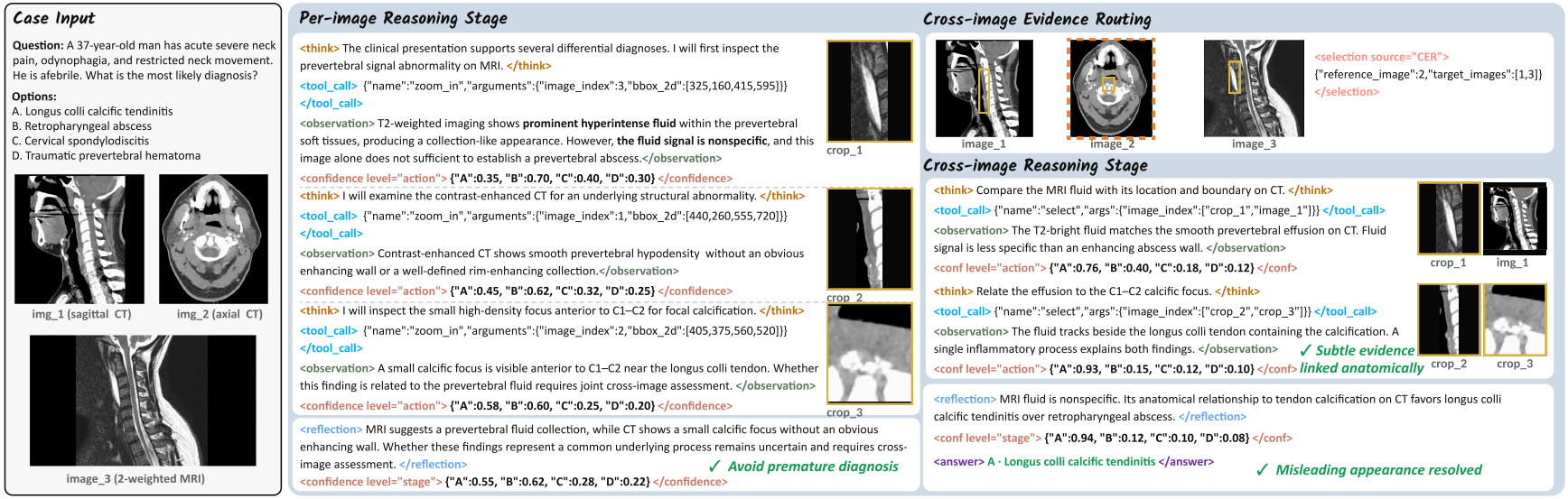


Figure 18: **Case 1: cross-image evidence revises an initial interpretation.** MRI shows prevertebral fluid, initially favoring an abscess. Comparing the corresponding CT appearance with the calcific focus anterior to C1–C2 supports longus colli calcific tendinitis (A). Gold boxes show inspected regions; the orange dashed outline identifies the CER reference image. Confidence values are model-reported candidate support, not calibrated diagnostic probabilities. Trace syntax is abbreviated for display.

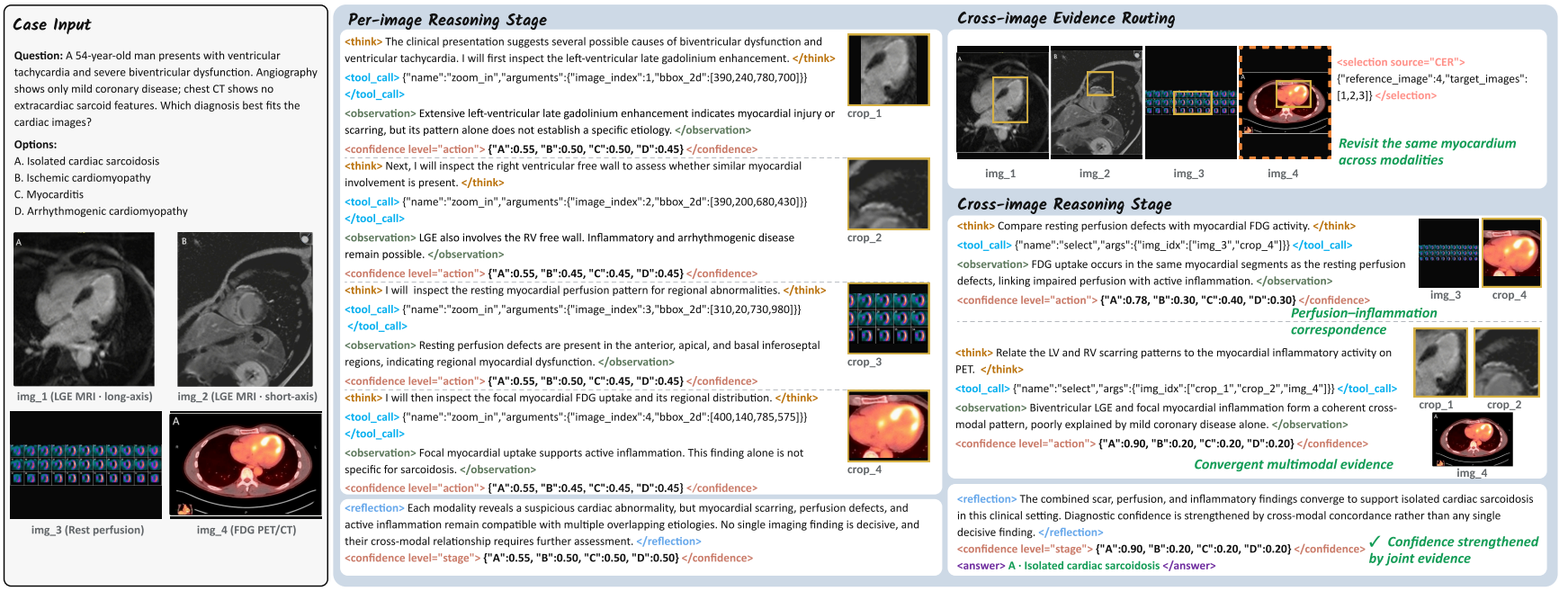


Figure 19: **Case 2: complementary modalities support a coherent diagnosis.** Joint inspection connects a perfusion defect with focal FDG uptake, then integrates these findings with biventricular MRI abnormalities. Together with the clinical context, the model selects isolated cardiac sarcoidosis (A). Gold boxes mark inspected regions; the orange dashed outline identifies the CER reference image. Green annotations summarize the cross-image relationships. Confidence readouts show model-reported support, and trace syntax is abbreviated for display. Figure 17 and the accompanying analysis explain why a temporary decrease in the truth-aligned margin can coexist with a positive direction.

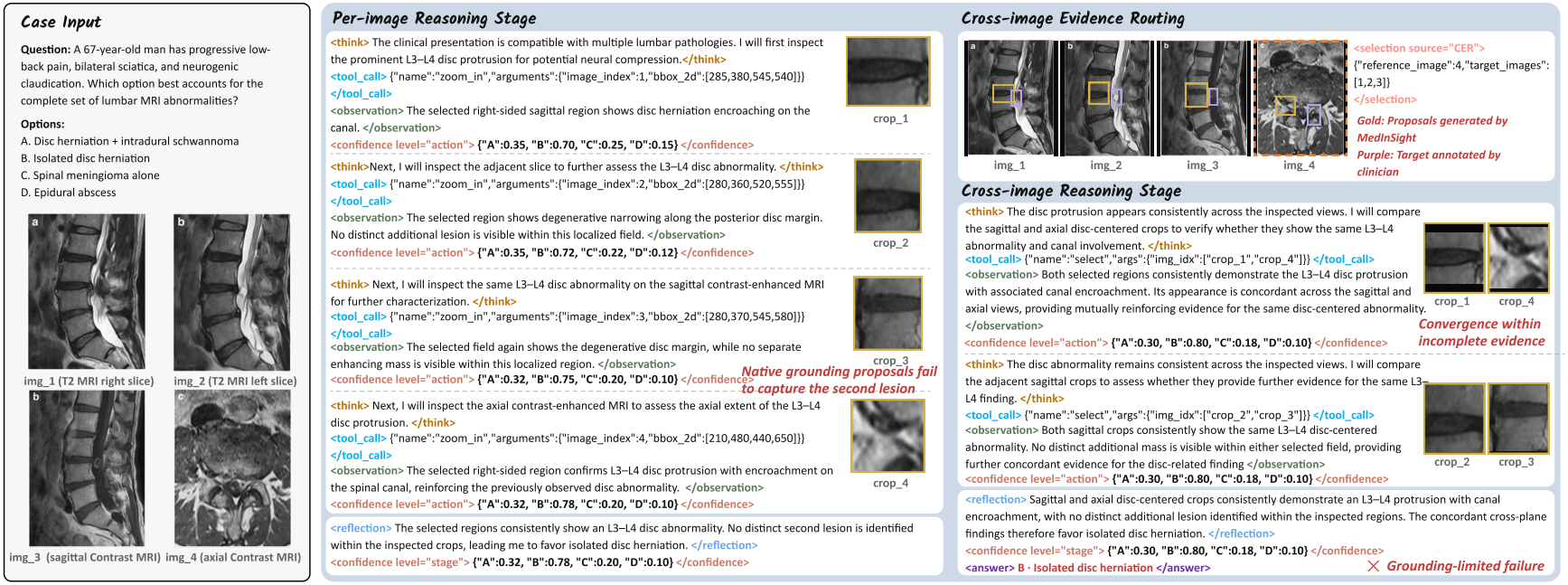


Figure 20: **Failure case: repeated agreement within an incomplete field of inspection.** Gold boxes show model-inspected disc regions, whereas purple boxes mark a second lesion in the clinician annotations. Cross-image reasoning reuses disc-centered crops and reinforces isolated disc herniation (B), overlooking the additional finding required by the question. CER makes all four images available, but availability alone does not ensure that the policy inspects all relevant evidence. This illustrates how restricted region coverage can sustain an incomplete diagnosis despite increasing model-reported confidence.

I ETHICS STATEMENT

This study uses public medical benchmarks and published clinical case reports. Our self-curated data are restricted to PMC Open Access reports under CC0 or CC BY licenses, as described in Section 3.4 and Appendix F.1. Published cases can still contain sensitive clinical information, and their reuse should respect source licensing and privacy constraints. The distribution of published cases and benchmark images may not reflect patient populations in practice. Missed findings, incorrect visual observations, and confident diagnostic errors remain possible. MedInSight is a research system evaluated retrospectively, not a validated tool for patient care or a substitute for qualified clinical judgment.

J REPRODUCIBILITY STATEMENT

We are committed to ensuring the reproducibility of our research. In the main paper and this appendix, we provide comprehensive details on the model architecture, training protocols, and hyperparameter settings. Our core mechanisms, Cross-image Evidence Routing (CER) and Cross-horizon Credit Assignment (CCA), are detailed in Section 3, which defines CER, CCA, and the training objective, and Algorithm 1, which summarizes the overall workflow. Appendix D reports model configurations, optimization settings, and baseline evaluation protocols. Appendix E specifies action interfaces, coordinate conventions, runtime feedback, action counting, and prompt templates. Appendix F documents data construction, filtering, held-out cases, and benchmark metrics. Extended comparisons and the CCA review protocol are provided in Appendix B. We plan to release the MedInSight source code and configuration files upon acceptance to facilitate full verification of our findings.