

# IBISAgent: Reinforcing Pixel-Level Visual Reasoning in MLLMs for Universal Biomedical Object Referring and Segmentation

## Supplementary Material

### A. Dataset Details

#### A.1. Dataset for SFT

Our dataset  $\mathcal{D}_{cold}$  for cold-start Supervised Fine-Tuning (SFT) is a large-scale, high-quality collection of textual reasoning and action trajectories for interactive segmentation. It comprises a total of 47,146 individual samples (e.g., slices or images), which collectively contain 456,795 visual question-answer (VQA) pairs that capture a wide range of diverse scenarios. The dataset is meticulously curated by filtering segmentation trajectories to ensure high fidelity, achieving an overall average Intersection over Union (IoU) of 94.27 (median: 95.07) and an average Dice score of 0.9703 (median: 97.47). This high level of accuracy confirms the quality of the segmentation ground truths generated by the trajectories. Furthermore, the average trajectory length per sample is 8.69 steps, indicating a rich capture of the multi-step refinement processes required for complex segmentation tasks.

##### A.1.1. Modality Diversity

The dataset is characterized by its extensive diversity, covering 9 distinct medical imaging modalities. This broad range ensures that models trained on this data can generalize across various imaging types, from common modalities like CT and MRI to more specialized ones like Pathology and OCT. A detailed breakdown of the dataset composition by modality is presented in Tab. 6.

##### A.1.2. Task Diversity

In addition to modality diversity, the dataset spans 38 distinct segmentation tasks. These tasks are derived from a combination of the Medical Segmentation Decathlon (MSD) dataset [36], covering both organs (e.g., ‘liver’, ‘heart’) and tumors (e.g., ‘brain tumor’), and various other specialized public biomedical datasets (e.g., ‘ACDC’, ‘LIDC-IDRI’, ‘GlaS’) derived from BioMedParse [46]. This task diversity exposes the model to a wide array of anatomical structures, pathologies, and image characteristics. The detailed statistics for each task group are provided in Sec. E.

### A.2. Dataset for RL

#### A.2.1. Pixel-Level Reasoning Corpus for RL.

In addition to the cold-start SFT dataset, we curated a large-scale, high-quality corpus specifically for the RL stage. As detailed in Sec. 3.3, this corpus is also sampled from BioMedParse [46] including some samples used in  $\mathcal{D}_{cold}$ . It

contains only the images, ground-truth masks, and QA pairs that require identifying fine-grained visual cues and localizing the specified mask region, thereby compelling the agent to autonomously explore and learn effective action policies.

This RL corpus comprises a total of 60,826 samples, which collectively contain 564,385 QA pairs. Similar to the SFT dataset, this corpus is highly diverse, spanning 8 medical imaging modalities and 39 distinct segmentation tasks. Detailed statistical breakdowns by modality and task group are provided in Fig. 4a and Fig. 15a, respectively.

#### A.2.2. Commonly-Used Medical VQA Datasets.

During RL training, we jointly use common medical VQA datasets and our curated pixel-level reasoning corpus. This hybrid training strategy preserves the model’s visual understanding and question-answering abilities, enabling the policy to selectively invoke pixel-space reasoning only when necessary. Tab. 7 reports the number of VQA pairs in the medical VQA datasets used in our training.

### A.3. Testing Benchmarks

#### A.3.1. In-domain test set $\mathcal{D}_{test}$

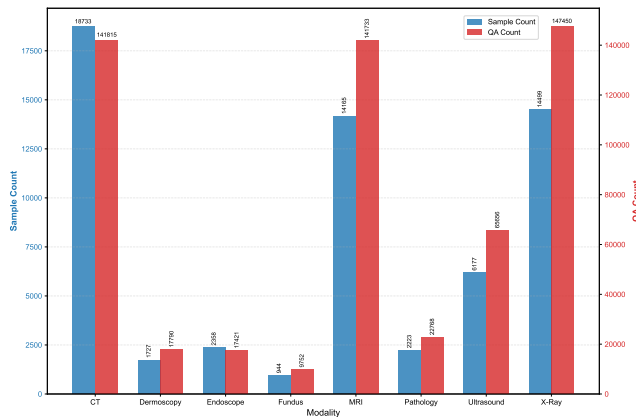
To comprehensively evaluate the model’s generalization capabilities and robustness in pixel-level reasoning, we constructed a diverse test set  $\mathcal{D}_{test}$  that is strictly disjoint from the training corpora ( $\mathcal{D}_{cold}$  and the  $\mathcal{D}_{rl}$  corpus). This dataset is designed to assess performance across a wide spectrum of medical imaging modalities and challenging segmentation scenarios.

The test set comprises a total of 9,902 samples, containing 156,289 VQA pairs. Unlike the training sets, which emphasize breadth by covering a wide variety of tasks, the test set focuses on more complex and challenging scenarios. Specifically, the test set places greater emphasis on fine-grained visual feature recognition—such as identifying tiny structures, intricate abnormalities, lesions, and tumors—to ensure that the benchmark rigorously evaluates the model’s ability to handle challenging targets. Consequently, the test set spans 32 distinct task groups.

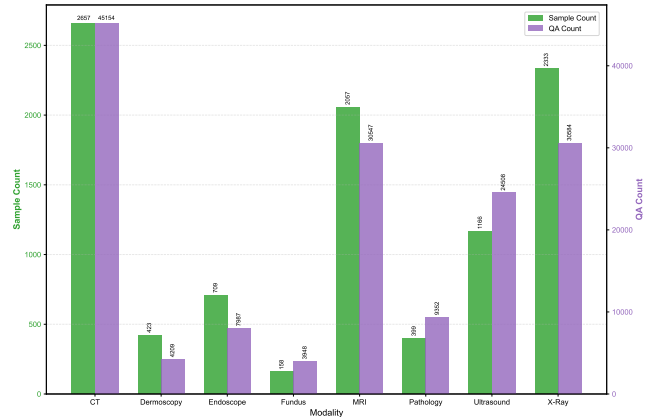
In terms of modality,  $\mathcal{D}_{test}$  covers 8 major medical imaging types: CT, MRI, X-Ray, Ultrasound, Pathology, Endoscopy, Dermoscopy, and Fundus. This distribution ensures that the evaluation reflects real-world clinical diversity. The statistical distributions of samples and QA pairs by modality and by specific dataset are visualized in Fig. 4b and Fig. 15b, respectively.

Table 6. Statistical overview of the **SFT** dataset  $\mathcal{D}_{cold}$ , categorized by imaging modality. Detailed breakdown by task group is shown in Sec. E.

Modality	Samples	Total QAs	Avg. Length	Avg. IoU	Median IoU	Avg. DSC	Median DSC
<b>CT</b>	22,504	209,291	8.30	0.9462	0.9519	0.9722	0.9753
<b>Dermoscopy</b>	1,302	13,086	9.05	0.9348	0.9338	0.9662	0.9658
<b>Endoscope</b>	2,431	17,434	6.17	0.9548	0.9609	0.9768	0.9800
<b>Fundus</b>	472	4,850	9.28	0.8834	0.8772	0.9378	0.9346
<b>MRI</b>	7,453	82,616	10.08	0.9404	0.9457	0.9691	0.9721
<b>OCT</b>	205	1,850	8.02	0.9116	0.9074	0.9537	0.9515
<b>Pathology</b>	1,195	11,765	8.85	0.9081	0.9243	0.9501	0.9607
<b>Ultrasound</b>	3,499	36,972	9.57	0.9341	0.9360	0.9659	0.9669
<b>X-Ray</b>	8,085	78,931	8.76	0.9460	0.9542	0.9721	0.9766
<b>Total</b>	<b>47,146</b>	<b>456,795</b>	<b>8.69</b>	<b>0.9427</b>	<b>0.9507</b>	<b>0.9703</b>	<b>0.9747</b>



(a) **RL Corpus  $\mathcal{D}_{rl}$** : Statistics by Modality



(b) **In-domain Test Set  $\mathcal{D}_{test}$** : Statistics by Modality

Figure 4. **Modality distribution.** (a) The RL corpus  $\mathcal{D}_{rl}$  (training stage) contains 60, 826 samples and 564, 385 QAs across 8 modalities. (b) The In-domain Test set  $\mathcal{D}_{test}$  comprises 9, 902 samples and 156, 289 QAs covering the same 8 modalities. The dual-axis plots show the sample count (left axis) and total QA pairs (right axis) for each category.

Table 7. The distribution of commonly-used medical VQA datasets used in RL stage. “HuatuoV\_A” and “HuatuoV\_I” refer to the Huatuo\_PubMedVision\_Alignment and Huatuo\_PubMedVision\_InstructionTuning VQA datasets, respectively.

Dataset	VQA Number
<b>HuatuoV_A</b>	128000
<b>HuatuoV_I</b>	128000
<b>PMC_VQA</b>	32000
<b>VQA_RAD</b>	8000
<b>SLAKE</b>	9000
<b>PATH_VQA</b>	19000

### A.3.2. Out-of-domain test set MeCOVQA-G+

Datasets for training and evaluating text-segmentation alignment in the medical domain are extremely scarce. One of the few publicly available resources is MeCoVQA-G, which was recently introduced alongside the MedPlib paper [13]. MeCoVQA-G is a large-scale, pixel-level VQA subset of the MeCoVQA family, constructed by pairing biomedical images with natural-language questions that explicitly ask the model to segment a given anatomical structure or lesion. Each sample contains a 2D image slice, a templated question targeting a specific anatomical class, and the corresponding binary segmentation mask as the ground-truth answer. The released split is 100K training pairs and 2, 344 test pairs.

MeCOVQA-G+ [40] is a thoroughly re-annotated and expanded edition of the MeCoVQA-G [13]. MeCOVQA-

G+ increases both the scale and modality diversity of its predecessor, delivering a more reliable and comprehensive benchmark for medical text-to-segmentation tasks. MeCOVQA-G+ comprises 3,157 carefully curated text-segmentation pairs. The samples span a wide range of modalities, including X-ray, CT, MRI, ultrasound, and endoscopy. Each image has been meticulously reviewed by a team of medical experts to ensure the accuracy of the segmentation masks.

### A.3.3. Held-out in-house test set

For testing, in addition to our in-domain test set  $\mathcal{D}_{\text{test}}$ , we use a completely held-out in-house dataset comprising 1k CT, MRI, and histopathology images across 7 cancer types for zero-shot evaluation. In this held-out set, CT images include 100 liver tumor cases, 100 gallbladder tumor cases, 100 pancreatic cancer cases, and 100 kidney tumor cases. MRI images include 100 colon tumor cases and 100 brain cancer cases. Histopathology images include 400 breast cancer cases. For each case, human annotators construct a VQA pair along with a corresponding reasoning trajectory.

## B. More Implementation Details

### B.1. Trajectory Generation

To train our model for multi-step medical image segmentation, we require a dataset of expert-like interaction trajectories. We employed an automated algorithm to generate these trajectories by simulating the sequential refinement process an expert annotator would perform, leveraging the click simulation strategy proposed in [42]. The algorithm iteratively interacts with a pre-trained interactive segmentation model (specifically MedSAM2 [27]), intelligently placing clicks to correct errors in the current prediction until it closely matches the ground truth.

The core of this method is a deterministic, greedy strategy for selecting the next interaction point, augmented by a **mask prompting mechanism** to ensure stability. At each step  $t$ , we strictly utilize the low-resolution mask logits from the previous step, denoted as  $M_{\text{logits}}^{(t-1)}$ , alongside the cumulative click history  $H_t$ . This simulates a realistic annotation workflow where the annotator refines an existing mask rather than starting from scratch at each interaction.

First, we identify the error regions in the current prediction  $M_p^{(t)}$ . The False Negative (FN) region,  $M_{fn}$ , represents the target area missed by the model, while the False Positive (FP) region,  $M_{fp}$ , represents areas incorrectly included in the prediction:

$$M_{fn} = M_{gt} \setminus M_p^{(t)}, \quad M_{fp} = M_p^{(t)} \setminus M_{gt} \quad (4)$$

To emulate human behavior prioritizing large error regions, we compute the Euclidean distance transform for both error masks, denoted as  $D_{fn}$  and  $D_{fp}$ . The algorithm

selects the next click  $a_t = (c_t, l_t)$  by targeting the pixel with the maximum distance value (i.e., the center of the largest error region):

$$(c_t, l_t) = \begin{cases} (\arg \max D_{fn}, 1) & \text{if } \max(D_{fn}) \geq \max(D_{fp}) \\ (\arg \max D_{fp}, 0) & \text{otherwise.} \end{cases} \quad (5)$$

Crucially, the update rule for the segmentation model  $\mathcal{S}$  incorporates both the updated history and the dense mask prompt from the previous iteration:

$$M_p^{(t+1)}, M_{\text{logits}}^{(t+1)} = \mathcal{S}(I, H_t \cup \{a_t\}, M_{\text{logits}}^{(t)}) \quad (6)$$

where  $M_{\text{logits}}^{(0)}$  is initialized as None. This iterative process continues until the IoU exceeds a threshold  $\tau_{iou}$  or the maximum step count  $T_{max}$  is reached. The implementation logic is detailed in Algorithm 1. An example process is depicted as Fig. 5.

### B.2. QA Generation

To train IBISAgent to understand diverse user intents and perform pixel-level tasks, we constructed a comprehensive instruction dataset. We utilized Gemini-2.5-Pro to generate a rich set of visual question-answering (VQA) pairs and instructions.

**Hallucination Prevention via Fact-Based Generation.** A critical challenge in generating medical instructions is preventing the LLM from hallucinating non-existent anatomical features or pathologies. To mitigate this, we strictly conditioned the generation process on ground-truth evidence. Specifically, we provided Gemini-2.5-Pro with the raw image, the ground-truth mask, and a verified caption of the biomedical object. The model was explicitly instructed to generate prompts **only** based on these visible facts, ensuring that every instruction (e.g., ‘‘Segment the left lung’’) corresponds to an object actually present in the image.

**Hierarchical Instruction Categories.** We designed a taxonomical prompt library to cover different phases of the segmentation process, as illustrated in Fig. 6:

- **Initialization Prompts.** These prompts initiate the segmentation task from scratch. To mimic real-world user behavior, we categorized them into 7 broad types, ranging from *Direct Commands* to *Goal-Oriented* statements.
- **Specialized Query Templates.** We applied a 70/30 split between standard imperative prompts and **Interrogative Queries** to enhance the model’s flexibility. For the latter, we designed 5 specific sub-templates to mimic clinical uncertainty:
  - (i) *Conditional Logic*: ‘‘Is there a {object\_name}? If so, please segment it.’’
  - (ii) *Compound Requests*: ‘‘Can you find and segment the {object\_name}?’’
  - (iii) *Clinical Protocol Tone*: ‘‘I need to verify the presence of a {object\_name}...’’

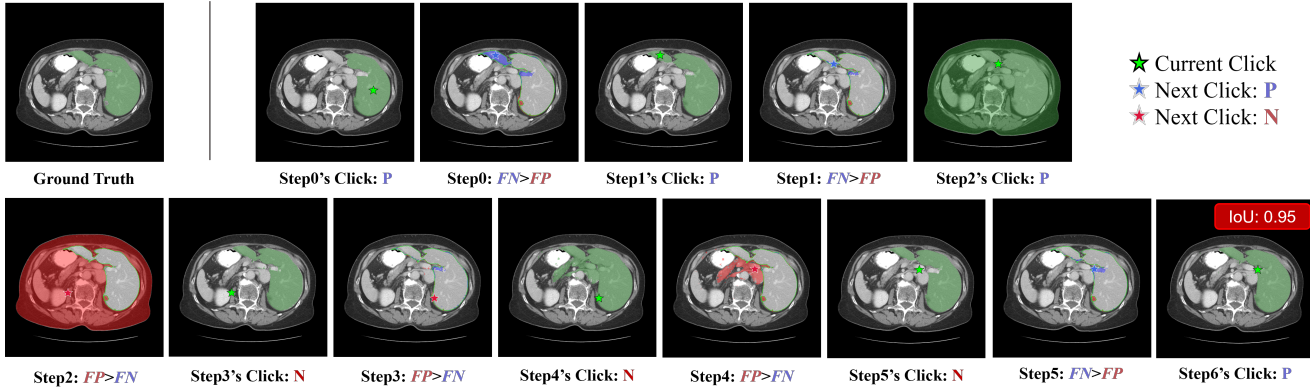


Figure 5. An illustrative example of the automated trajectory generation process for liver segmentation. The algorithm progressively refines the predicted mask through iterative interactions.

For each iteration (e.g., Step 0), two visualization panels are presented: (1) The **first image** displays the *current segmentation state*, showing the predicted mask (green translucent overlay) generated by the current click (marked by a green star). (2) The **second image** illustrates the *error analysis* against the Ground Truth (delineated by a **green outline**). The differences are visualized as blue translucent regions for False Negatives (FN, under-segmentation) and red translucent regions for False Positives (FP, over-segmentation). The star in this panel indicates the calculated **next action** based on the largest error region: a **blue star** denotes a Positive Click (P) to correct under-segmentation, while a **red star** denotes a Negative Click (N) to correct over-segmentation.

(iv) *Indirect/Conversational*: “I’m wondering if there’s a {object\_name}...”

(v) *Concise Checks*: “Visible {object\_name}? Please provide segmentation.”

- **Refinement Prompts.** We categorized them into 6 types. These prompts are used during the iterative interaction steps. They focus on fine-grained adjustments, such as *Requesting Next Steps* (e.g., “What is the next step?”), *Error Correction* (e.g., “The mask extends beyond the boundary”), and *Verification* (e.g., “Is this segmentation complete?”).

**Diverse Assistant Response Generation.** To ensure that the agent’s output is naturalistic and varied rather than robotic, we also constructed a template library for the **Assistant’s final responses**. These are categorized into 5 styles: *Direct & Concise* (“Segmentation complete.”), *Confident Affirmation* (“The object has been successfully segmented.”), *Object-Referencing* (“The {object\_name} is fully segmented.”), *Question-Answering* (“Yes, the object was found...”), and *Conversational* (“All done!”). This diversity prevents the model from overfitting to a single termination phrase.

During data construction, we dynamically fill all user and assistant templates with the specific anatomical target name (e.g., “left ventricle”) and imaging modality (e.g., “MRI”), ensuring high relevance and grammatical correctness.

### B.3. Reasoning Generation for Our SFT Dataset

A cornerstone of our SFT dataset is the high-quality, step-by-step reasoning (<think>...</think>) that accom-

panies each agent action. Generating this data presents a significant challenge: our agent (M1) operates from a limited, first-person perspective (seeing only the current segmentation mask), but the optimal reasoning for its next action (e.g., “correcting an over-segmentation”) requires an “oracle” or “ground-truth” perspective (knowing the precise False Positive and False Negative regions).

To solve this, we employed a “Teacher-Student” (or “Oracle-Agent”) generation pipeline. We utilized the powerful **GPT-5** model as the “Teacher” (M2) to synthesize reasoning traces for our “Student” agent (M1).

Our core innovation lies in a sophisticated prompt strategy that leverages the advanced role-playing capabilities of GPT-5 to bridge the information gap between the Oracle and the Agent.

1. **Persona and Perspective Simulation.** The system prompt instructs M2 to adopt the persona of an “expert radiologist.” Critically, it commands M2 to generate reasoning strictly from the limited perspective of the junior agent. The prompt explicitly states: “*This agent ONLY SEES ONE THING: a single, combined green mask... Write as if you are genuinely deducing the next step from only the visible image.*” This forces the teacher model to reverse-engineer the logic: instead of simply stating the error (which it knows), it must explain **why** the visual features (e.g., texture differences, anatomical landmarks) suggest an error exists.

2. **Privileged Information.** While M2 writes from the agent’s perspective, it views a privileged “oracle” image. As shown in Fig. 7 (Left), this image explicitly visualizes segmentation errors: a **Green Mask** for True Positives, a **Red Mask** for False Positives (over-segmentation), and a

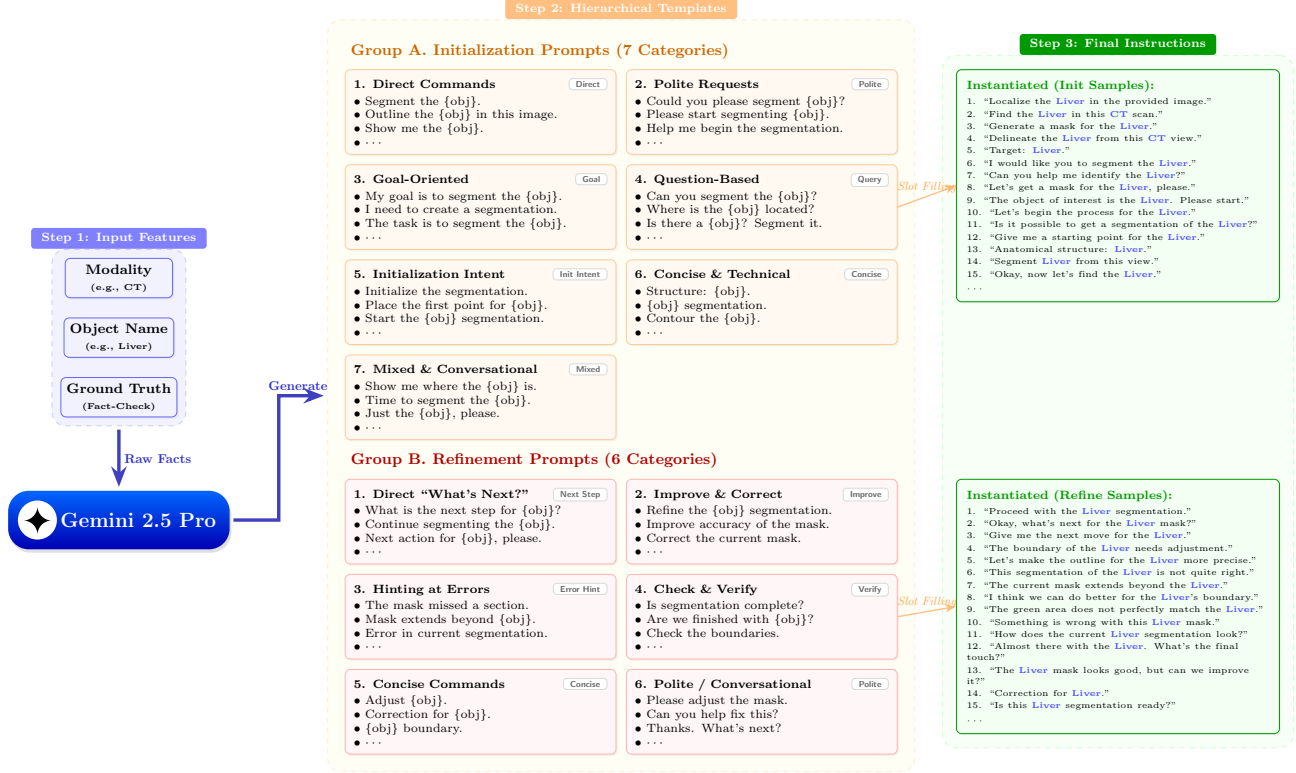


Figure 6. **The hierarchical prompt generation pipeline.** To ensure both diversity and factual accuracy, we leverage **Gemini 2.5 Pro** to synthesize a comprehensive instruction library. Conditioned on ground-truth input features (Step 1), the model dynamically generates a taxonomical prompt set (Step 2) divided into **Group A: Initialization Prompts** (7 categories, covering imperative to conversational tones) and **Group B: Refinement Prompts** (6 categories, focusing on iterative corrections). These templates are then instantiated into final instructions (Step 3), creating a rich dataset (50 items per group) that covers diverse user intents while strictly adhering to visual facts.

**Blue Mask** for False Negatives (under-segmentation). It also indicates the “correct” next action (Positive/Negative point).

3. **Preventing Information Leakage.** A primary risk in this pipeline is “prompt leakage,” where the teacher accidentally reveals its privileged knowledge (e.g., mentioning “the red mask”). Thanks to the superior instruction-following capability of GPT-5 compared to smaller models, we effectively mitigated this using a robust set of **Forbidden Concepts**. The system prompt strictly prohibits the output from containing terms like ‘Red’, ‘Blue’, ‘Cross’, ‘TP’, ‘FP’, ‘FN’, or ‘Ground Truth’.

As shown in Fig. 8, this approach generates dense, anatomically grounded reasoning traces without requiring manual templates for every scenario (like we did before using smaller models).

## C. Reward Functions

**The reasoning-format reward.** The reasoning-format reward  $\mathcal{S}_{format}$  evaluates the structural validity of  $R$  by ver-

ifying that the model’s output includes all required special tokens in the prescribed order. Specifically, the model should enclose its chain-of-thought between `<think>` and `</think>` tags, place the tool-call choices and parameters between `<tool_call>` and `</tool_call>` tags, and place the final answer between `<answer>` and `</answer>` tags. Outputs that adhere to this structure receive a positive reward.

$$\mathcal{S}_{format} = \begin{cases} 1, & \text{if all required fields appear} \\ & \text{and are in the correct order,} \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

**The final-answer reward.** The final-answer reward  $\mathcal{S}_{ans}$  encompasses multiple task types, thereby providing the agent with diverse feedback. For **multiple-choice questions**, we simply check the exact match between the predicted answer and the ground truth:

$$\mathcal{S}_{ans}(A, \hat{A}) = \mathbb{I}(A = \hat{A}). \quad (8)$$

---

**Algorithm 1:** Pseudo-code for Trajectory Generation
 

---

```

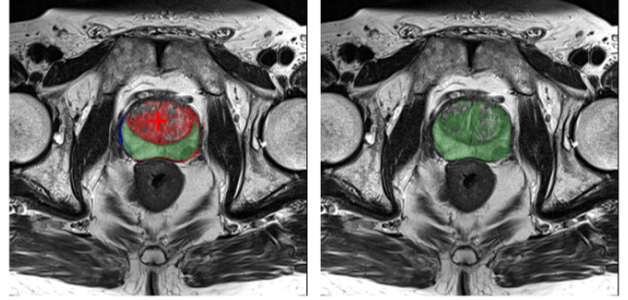
1 def generate_trajectory(image, gt_mask, model,
2   max_steps=20, iou_thresh=0.95):
3   pred_mask = np.zeros_like(gt_mask)
4   low_res_logits = None
5   history = []
6
7   for step in range(max_steps):
8     # Check Stop Condition
9     iou = calculate_iou(pred_mask, gt_mask)
10    if iou >= iou_thresh:
11      break
12
13    # Identify Error Regions (FN & FP) and
14    Find Click
15    fn_mask = (gt_mask == 1) & (pred_mask ==
16    0)
17    fp_mask = (gt_mask == 0) & (pred_mask ==
18    1)
19
20    fn_dist = cv2.distanceTransform(fn_mask.
21    astype(np.uint8), ...)
22    fp_dist = cv2.distanceTransform(fp_mask.
23    astype(np.uint8), ...)
24
25    if np.max(fn_dist) >= np.max(fp_dist):
26      coords = np.unravel_index(np.argmax(
27      fn_dist), fn_dist.shape)
28      history.append((coords, 1))
29    else:
30      coords = np.unravel_index(np.argmax(
31      fp_dist), fp_dist.shape)
32      history.append((coords, 0))
33
34    # Update Prediction using Points AND
35    Previous Mask Logits
36    points = [p[0] for p in history]
37    labels = [p[1] for p in history]
38
39    pred_masks, scores, logits = model.
40    predict(
41      point_coords=points,
42      point_labels=labels,
43      mask_input=low_res_logits
44    )
45
46    # Select best mask and update logits for
47    next iteration
48    best_idx = np.argmax(scores)
49    pred_mask = pred_masks[best_idx]
50    low_res_logits = logits[best_idx]
51
52    return history, pred_mask

```

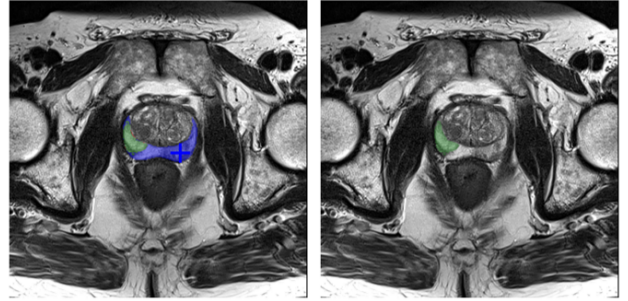
---

Here,  $A$  denotes the ground-truth answer and  $\hat{A}$  is the predicted answer obtained by rule-based parsing of the model’s final output. The indicator  $\mathbb{I}$  is defined to be 1 if  $A = \hat{A}$  and 0 otherwise.

For **segmentation** tasks, in contrast to earlier reward designs, we use MedSAM2 as external reward providers.



(a) A Negative Click example for Prostate segmentation. The M2 (oracle) view (Left) shows the False Positive (FP, red) region. The M1 (agent) view (Right) only sees the combined green mask.



(b) A Positive Click example for Prostate segmentation. The M2 (oracle) view (Left) shows the False Negative (FN, blue) region. The M1 (agent) view (Right) only sees the incomplete green mask.

**Figure 7. Comparison of the “Oracle” view (for M2) and the “Agent” view (for M1) used in SFT reasoning generation.** Notably, this agent-visible mask (green) is the sum of the oracle’s True Positive (green) and False Positive (red) areas. M1 must learn to infer the expert’s corrective reasoning from this limited perspective.

Given either click points predicted by the MLLM, we query MedSAM2 to obtain a segmentation mask. We then compute the IOU between this mask and the ground-truth mask and assign piecewise rewards as follows:

$$\mathcal{S}_{\text{ans}} = \begin{cases} 3, & \text{IoU} > 0.80, \\ 2, & 0.70 < \text{IoU} \leq 0.80, \\ 1, & 0.50 < \text{IoU} \leq 0.70, \\ 0, & \text{otherwise.} \end{cases} \quad (9)$$

This segmentation reward supplies strong positive feedback only when the predicted region closely matches the ground truth, while at lower IoU levels it provides guidance that encourages gradual improvement.

**Region-based click placement reward.** A core innovation of our framework is that we introduce explicit spatial constraints into the reward signal. Unlike generic RL agents that may learn to click arbitrarily to trigger tool usage, we enforce semantically valid interactions through  $\mathcal{S}_{\text{click}}$ . This reward serves as a dense supervision signal, ensuring that the agent’s actions are grounded in the anatomical reality of

### SYSTEM PROMPT (Persona & Constraints)

You are an expert radiologist. Your task is to examine a user's prompt, a ground-truth image, and a given correct action, and then generate a single, high-quality <think> block with the detailed, analytical reasoning that justifies that specific action, all from the perspective of a segmentation agent.

#### 1. Your Core Task and Persona

You must adopt the persona of an agent that only sees the medical image and a plain green mask (if present). You will be secretly shown a ground-truth image to guide your reasoning, but you must never reveal this secret knowledge.

##### • Secret Knowledge (Teacher View):

Green/Red/Blue Masks: Indicate correctly, over-, and under-segmented areas.  
Blue/Red Crosses (+): Indicate the correct location for a Positive/Negative point.-

##### • Agent's View (Your Persona):

Sees only a uniform green mask which is the sum of the correct (Green) and over-segmented (Red) areas.

#### 2. Reasoning Instructions for Each Scenario

Your <think> block must reverse-engineer the correct action from the agent's limited perspective, using your expert anatomical knowledge.

##### (1) For Initialization (No Mask):

You see a Blue Cross (+). Justify why this location is an ideal starting point. Analyze its anatomical position (e.g., center of the target, clear texture, away from ambiguous borders).

##### (2) For Refinement (Under-segmentation):

You see a Blue Mask (missed area) and a Blue Cross (+). The agent sees an incomplete mask. Your reasoning must identify the

missing anatomical part and explain why a positive point is needed there.

Example logic: "The current mask covers the main body of the liver, but it's missing the superior lobe. To complete the segmentation, I must add a point to this unmasked region."

##### (3) For Refinement (Over-segmentation):

You see a Red Mask (extra area) and a Red Cross (+). The agent sees a mask that is too large. Your reasoning must identify which adjacent structure has been incorrectly included and explain why a negative point is necessary.

Example logic: "The mask for the right kidney appears to have bled into the adjacent psoas muscle. To correct this, I will place a negative point on the incorrectly included muscle tissue."

##### (4) For Termination (No Errors):

You see no error masks or crosses. The agent sees a green mask. Your reasoning must be a step-by-step evaluation of the mask's boundaries against known anatomical landmarks. Conclude that the segmentation is highly accurate and optimally aligned. You should reason that any remaining minor imperfections are clinically insignificant and cannot be substantially improved with another single point, thus justifying termination.

#### 3. Mandatory Rules

(1) **FORBIDDEN CONCEPTS:** Your output inside <think> MUST NOT contain any of the following words or refer to their concepts: Red, Blue, Cross, TP, FP, FN, Ground Truth, Teacher, Junior, Correct. Adherence is critical.

(2) **MAINTAIN PERSONA:** Write as if you are genuinely deducing the next step from only the visible image and the plain green mask. Never reveal your access to the secret knowledge.

(3) **STRICT FORMAT:** Your entire response must be only the <think>...</think> block and its content. No other text, tags, or formatting.

(4) **EXPERT QUALITY:** Provide a detailed, step-by-step thought process that reflects deep anatomical understanding. Avoid short, superficial statements.

### USER PROMPT (Context Injection)

Here is the context for the current segmentation step.

--- USER INSTRUCTION ---

The mask for the prostate has missed a section.

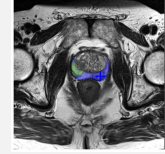
--- CORRECT ACTION ---

Positive Click (0.5325, 0.6794)

--- YOUR TASK ---

Based on the provided ground-truth image, generate the detailed <think> block that justifies the specified correct action from the perspective of the segmentation agent.

### VISUAL INPUT (Privileged View)



Teacher Model

### GENERATED THINKING (Student Logic)

The current mask identifies the central gland of the prostate but fails to cover the peripheral zone on the right lateral side. Examining the T2-weighted intensity, the unmasked region shares the same hyperintense texture as the segmented area, bounded by the clear hypointense line of the prostatic capsule which lies further out. The current boundary stops prematurely before reaching this anatomical landmark, resulting in an under-segmentation of the organ. To complete the segmentation, a positive point should be placed in this missed lateral region to expand the mask outward until it aligns with the true anatomical capsule.

Figure 8. **The prompt engineering pipeline for synthesizing pixel-level reasoning traces.** To bridge the gap between ground-truth knowledge and the agent's limited perspective, we employ a Teacher-Student strategy using GPT-5. The pipeline integrates three key components: (1) a System Prompt that establishes an expert persona and enforces strict constraints; (2) the User Context containing the target instruction and correct action; and (3) the Privileged Oracle View where error regions are explicitly visualized. The model is tasked with "reverse-engineering" the reasoning, producing vision-based justifications for the corrective action without revealing its access to the privileged information.

the image.

Let  $a_t = (c_t, p_t)$  denote the action at step  $t$ , where  $c_t \in \mathbb{R}^2$  is the spatial coordinate and  $p_t \in \{+1, -1\}$  indicates a positive or negative click type. Let  $M_{t-1}$  be the segmentation mask from the previous step (with  $M_0$  initialized as an empty mask). We define the eligible error regions for interaction based on the ground truth mask  $M_{gt}$ :

$$\Omega_{FN} = M_{gt} \setminus M_{t-1}, \quad \Omega_{FP} = M_{t-1} \setminus M_{gt} \quad (10)$$

A click is considered valid if a positive point falls within the under-segmented region ( $\Omega_{FN}$ ) or a negative point falls within the over-segmented region ( $\Omega_{FP}$ ). The reward function is formalized as:

$$S_{\text{click}}(a_t) = \begin{cases} r_{\text{click}}, & \text{if } p_t = +1 \wedge c_t \in \Omega_{FN}, \\ r_{\text{click}}, & \text{if } p_t = -1 \wedge c_t \in \Omega_{FP}, \\ -\lambda_{\text{miss}}, & \text{otherwise.} \end{cases} \quad (11)$$

where  $r_{\text{click}} = 1$  is a positive bonus for spatially accurate clicks, and  $\lambda_{\text{miss}} = 1$  is a penalty for invalid clicks. This reward effectively guides the policy to minimize the symmetric difference between the predicted and ground-truth masks step-by-step.

**Progressive segmentation improvement reward.** To prevent the agent from engaging in redundant operations or oscillating between states without improving the result, we incorporate a progressive improvement reward  $S_{\text{pseg}}$ . This component evaluates the marginal contribution of each action to the overall segmentation quality.

Let  $\text{IoU}(M, M_{gt})$  denote the Intersection-over-Union between a mask  $M$  and the ground truth. We calculate the quality gain  $\Delta Q_t$  after executing action  $a_t$ :

$$\Delta Q_t = \text{IoU}(M_t, M_{gt}) - \text{IoU}(M_{t-1}, M_{gt}). \quad (12)$$

The reward is assigned only if the action yields a strictly positive gain:

$$S_{\text{pseg}} = \mathbb{I}(\Delta Q_t > 0) \quad (13)$$

This incentivizes the agent to strictly ascend the gradient of segmentation quality.

**Trajectory length reward.** Efficiency is a critical metric for clinical assistants. To encourage the model to achieve high-quality segmentation with the minimum number of interactions, we introduce a trajectory length reward  $S_{\text{len}}$ . Let  $T$  be the total number of steps taken in a reasoning path,

Methods	VQA-RAD	PathVQA	SLAKE	AVG.
GPT-4o [15]	64.9	58.1	70.9	64.6
LLaVA-Med-7B [21]	53.1	44.2	47.5	48.3
HuatuoGPT-Vision-34B [4]	62.0	51.3	69.5	60.9
Lingshu [43]	66.1	68.7	78.0	70.9
Chiron [37]	72.7	68.9	77.3	73.0
<b>IBISAgent</b>	<b>73.4</b>	<b>69.2</b>	<b>83.5</b>	<b>75.4</b>

Table 8. Comparison of IBISAgent with existing MLLMs on different VQA benchmarks.

and  $T_{\text{opt}}$  be a predefined optimal threshold. The reward is defined as:

$$S_{\text{len}} = \begin{cases} r_{\text{eff}}, & \text{if } T \leq T_{\text{opt}}, \\ -\gamma \cdot (T - T_{\text{opt}}), & \text{if } T > T_{\text{opt}}. \end{cases} \quad (14)$$

where  $r_{\text{eff}} = 1$  is a bonus for efficient completion, and  $\gamma = 0.2$  is a decay factor that applies a linear penalty for each additional step beyond the threshold. This formulation balances the trade-off between exhaustive refinement and interaction efficiency.

## D. More Experiments

### D.1. VQA Performance.

We also conducted experiments demonstrating that pixel-level reasoning not only improves segmentation performance but also enhances the model’s VQA capabilities. The results of evaluation across three medical VQA benchmarks are summarized in Tab. 8. IBISAgent outperforms both open-source and proprietary MLLMs. IBISAgent outperforms both open-source and proprietary MLLMs. Notably, compared with existing medical MLLMs trained on large-scale VQA datasets, IBISAgent achieves at least a 2.4% improvement in average accuracy. This further validates our motivation that enhancing MLLMs’ understanding of fine-grained medical image features fundamentally improves their medical image analysis capabilities. Pixel-level exploration of localized regions closely mirrors the way clinicians interpret and reason about medical images, and our work effectively stimulates and strengthens this critical capability in MLLMs.

### D.2. Impact of Segmentation Tool Types

We further examine the effect of replacing the interactive segmentation tool used by IBISAgent. Specifically, we substitute the default MedSAM 2 [27] with alternative tools, including MedSAM [26], SAM [17], and SAM 2 [31], and compare the resulting performance, as shown in Tab. 9. We find that IBISAgent remains highly robust to the choice of interactive segmentation tool, exhibiting only minor differences in final segmentation accuracy across these replacements. This robustness arises from IBISAgent’s ability to

Methods	MeCOVQA-G+			In-House Test set		
	IoU $\uparrow$	DSC $\uparrow$	F1 $\uparrow$	IoU $\uparrow$	DSC $\uparrow$	F1 $\uparrow$
IBISAgent (SAM)	79.95	88.74	94.83	71.82	83.19	90.98
IBISAgent (SAM 2)	80.32	89.01	95.08	71.93	83.40	91.36
IBISAgent (MedSAM)	80.29	89.00	95.03	71.91	83.37	91.32
<b>IBISAgent (MedSAM 2)</b>	<b>80.63</b>	<b>89.27</b>	<b>95.24</b>	<b>72.09</b>	<b>83.78</b>	<b>91.76</b>

Table 9. Ablation study on segmentation tool types.

generate precise click-point locations and perform multi-round refinement, which jointly help maintain segmentation quality and ensure strong test-time stability. These results indicate that IBISAgent can effectively adapt to a wide range of interactive segmentation tools, rather than relying solely on MedSAM 2.

### D.3. The Performance of the Segmentation Tools

We also report the standalone performance of the segmentation tools on the test sets as a reference, further highlighting the superiority of our method. Tab. 10 presents the results. In this comparison, we evaluate segmentation performance both with and without GT bounding-box prompts. Because MedSAM and MedSAM 2 support only visual prompts, their results in the “w/o bbox” setting are marked as  $\times$ . From Tab. 10, we observe that IBISAgent consistently achieves the highest performance across both segmentation modes. These findings indicate that IBISAgent exhibits strong generalization in text-driven segmentation and, when using GT bbox as the first step, delivers segmentation quality that consistently surpasses the competing tools such as MedSAM, MedSAM 2, and BiomedParse.

Overall, these results further validate the advantages of IBISAgent. Under our formulation, the agent performs precise pixel-level visual reasoning to accurately localize target regions and iteratively refine masks, enabling segmentation performance that exceeds the inherent upper bound of the underlying segmentation tools themselves.

### D.4. More Case Studies

We provide additional qualitative comparisons to further prove the robustness of IBISAgent in different anatomical regions. As illustrated in Fig. 9 and Fig. 10, we present two challenging scenarios including a low-contrast pancreatic tumor and an irregular lung tumor, respectively. Consistent with our observations in Sec. 4.2, existing MLLMs frequently suffer from severe hallucinations, incorrect grounding, or missed diagnoses when discerning subtle pathological cues. In contrast, IBISAgent successfully initiates correct segmentation and employs its unique reasoning-driven refinement mechanism to correct errors—such as retracting masks from adjacent healthy tissues—ultimately achieving high-quality segmentation and accurate diagnostic descriptions.

Models	In-domain test set				MeCOVQA-G+				In-House Test set			
	w/o bbox		with bbox		w/o bbox		with bbox		w/o bbox		with bbox	
	IOU	DSC	IOU	DSC	IOU	DSC	IOU	DSC	IOU	DSC	IOU	DSC
SAM 2 [26]	×	×	80.30	85.61	×	×	65.12	75.54	×	×	51.08	56.65
MedSAM [26]	×	×	79.73	85.44	×	×	60.32	71.74	×	×	49.50	53.28
MedSAM2 [27]	×	×	82.07	87.28	×	×	<u>71.30</u>	<u>81.12</u>	×	×	<u>59.24</u>	<u>64.49</u>
BiomedParse [46]	83.03	87.19	<u>84.28</u>	<u>89.67</u>	37.68	45.39	67.41	78.36	27.48	34.23	55.87	60.68
<b>IBISAgent</b>	<b>85.58</b>	<b>92.21</b>	<b>86.37</b>	<b>92.48</b>	<b>80.63</b>	<b>89.27</b>	<b>81.56</b>	<b>90.11</b>	<b>72.09</b>	<b>83.78</b>	<b>72.96</b>	<b>84.83</b>

Table 10. Comparison with interactive segmentation tools. Best and second-best results are shown in **bold** and underline, respectively. × means that the model does not support text instruction following.

We also present additional segmentation results in Fig. 11, showcasing IBISAgent’s multi-round segmentation trajectories on various biomedical images and illustrating its iterative refinement process across different segmentation tasks.

#### D.5. How IBISAgent Corrects Errors

In Fig. 12, we present several representative examples that illustrate how IBISAgent corrects different types of errors, further demonstrating its robustness. We consider three typical scenarios. (1) Deceptive or incorrect instructions: When the user provides misleading or erroneous instructions describing a nonexistent target, IBISAgent correctly recognizes that the specified object is not present in the image and refrains from producing an incorrect mask, highlighting its genuine understanding of fine-grained visual cues. (2) Inconsistent initial masks: During mask refinement, if the user supplies an initial mask that does not match the described segmentation target, IBISAgent detects the inconsistency, corrects the erroneous mask, and generates the appropriate segmentation result. (3) Backtracking to undo incorrect decisions: During multi-round mask refinement, IBISAgent has access to the full interaction history, enabling it to assess whether the current segmentation trajectory is reasonable. When an incorrect action leads to a sub-optimal mask, the agent can automatically backtrack, undo the erroneous decision, and re-plan its click sequence. This ability largely stems from our use of Reflective Behavior Synthesis during SFT, which provides synthetic examples of such behavior and strengthens the agent’s robustness.

#### D.6. RL Training Dynamics

To analyze the training dynamics, we plot the IoU reward against training steps in Fig. 13. The curve illustrates the overall improvement in segmentation performance throughout the RL process. As observed, the IoU reward exhibits a steady increase, which demonstrates the stability of the training procedure. Under the guidance of our designed reward, the agent continuously explores the environment, thereby progressively acquiring enhanced planning and seg-

mentation capabilities.

### E. System and User Prompts

The detailed system prompt and user prompt used by IBISAgent are shown as Fig. 14.

### F. Future Works

IBISAgent endows MLLMs with powerful pixel-level visual reasoning capabilities. Together with our novel behavioral formulation and training framework, IBISAgent substantially pushes the boundary of MLLM-based biomedical image reasoning and segmentation. Nevertheless, several open challenges remain. First, the current agent operates primarily in 2D settings; extending IBISAgent to 3D scenarios—or even developing a unified 2D–3D MLLM agent—represents a highly promising research direction. Second, further improvements in the efficiency of multi-step agentic interaction will be essential to reduce computational overhead.

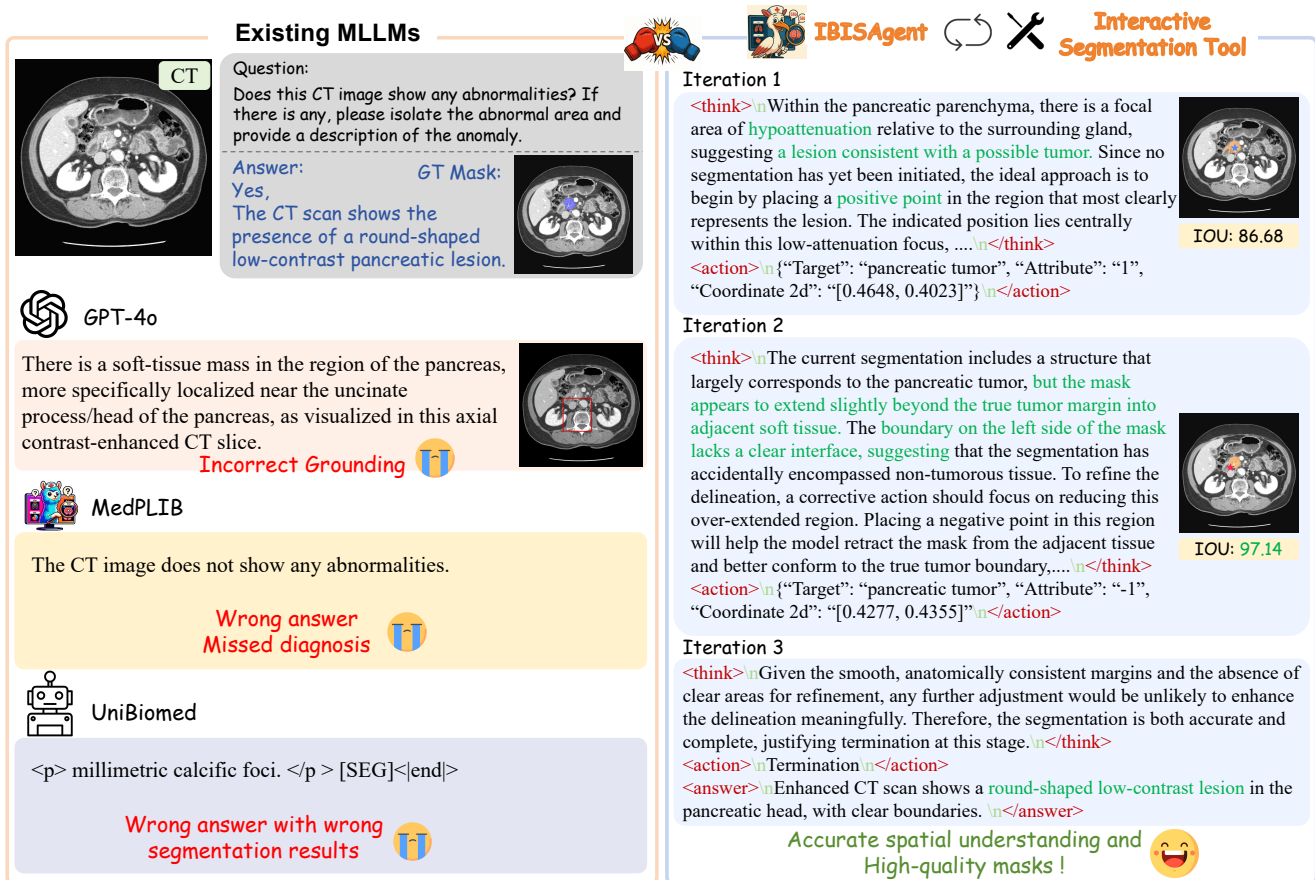


Figure 9. **Qualitative comparison on the pancreatic tumor case.** Existing MLLMs fail to provide reliable analysis: GPT-4o identifies the wrong location, MedPLIB misses the diagnosis entirely, and UniBiomed hallucinates unrelated calcific foci with an incorrect mask. Conversely, IBISAgent accurately identifies the low-contrast lesion and performs multi-step refinement to distinguish the tumor from the surrounding pancreatic parenchyma, achieving an IoU of 97.14%.

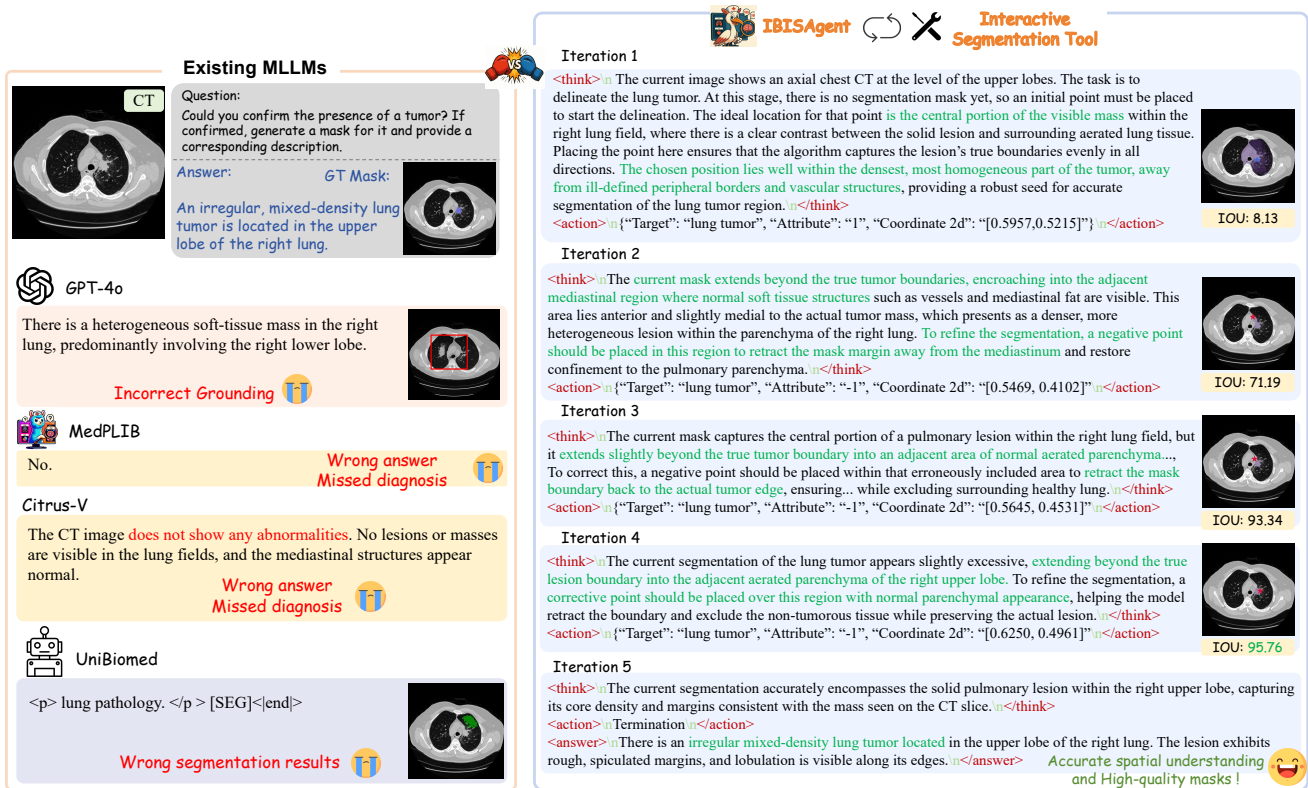


Figure 10. **Qualitative comparison on the lung tumor case.** While baseline models fail to detect the abnormality and GPT-4o mislocalizes the lesion, IBISAgent demonstrates superior pixel-level reasoning. Through five iterations, the agent detects the irregular mass and progressively corrects over-segmentation errors in the mediastinal and parenchymal regions, resulting in a 95.76% IoU.

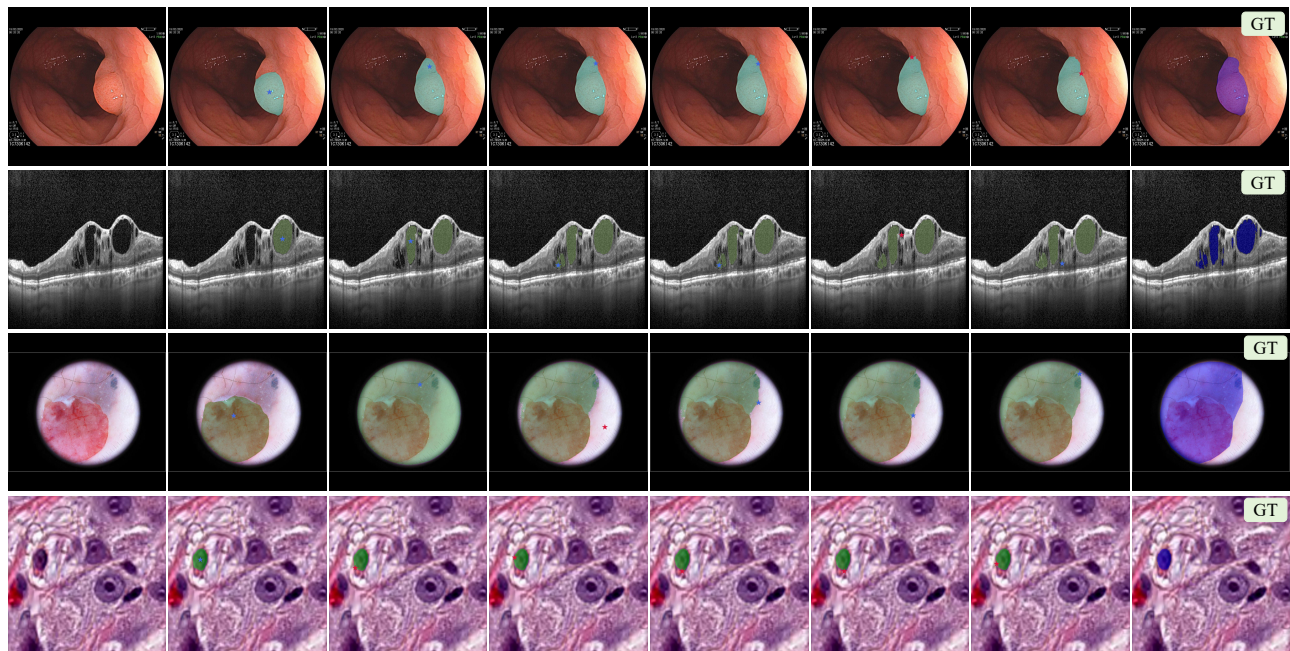


Figure 11. Visualization of IBISAgent’s multi-round segmentation trajectories on various biomedical images, illustrating its iterative refinement process across different segmentation tasks.

**User**

Question: There is a lung tumor in this image. Could you please help me segment it?

**IBISAgent**

This CT image appears to be an abdominal scan. The lungs are not visible in this slice, so it would not be possible to detect or evaluate a lung tumor in this image.

(1) How IBISAgent Handles Deceptive or Incorrect Instructions 😊

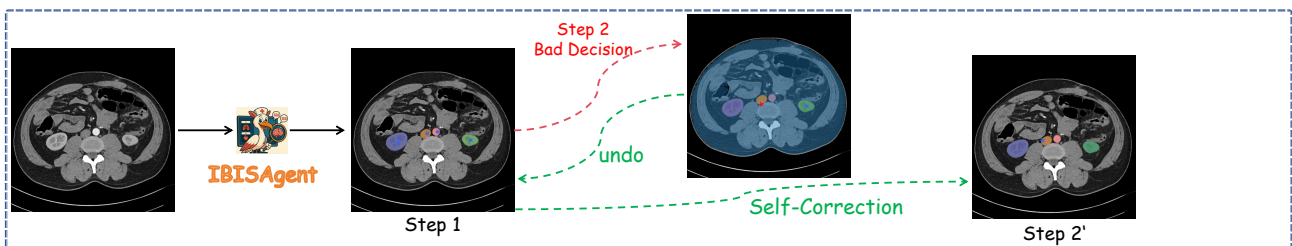
**User**

Question: User query: I am annotating the liver tumor region in this CT scan. The green area shows the part I have already labeled. Could you help me complete the remaining segmentation?

**IBISAgent**

The task is to continue refining the segmentation of the liver tumor. However, the user-provided annotation does not correspond to the liver tumor region, suggesting that the initial mask may be incorrect. I will therefore re-segment the liver tumor. As the first step, I will place a negative point to remove the erroneous initial mask. ....

(2) How IBISAgent Handles Incorrect User-Provided Annotation 😊



(3) An Instance of IBISAgent's Self-Correction Behavior 😊

Figure 12. Illustrations of how IBISAgent corrects different types of errors. (1) An example showing IBISAgent's response when the user provides deceptive or incorrect instructions describing a nonexistent target. (2) A case where the initial mask provided by the user does not match the described segmentation target, and how IBISAgent reacts accordingly. (3) An example demonstrating IBISAgent's ability to backtrack and undo an incorrect decision, followed by re-planning and selecting new click points.

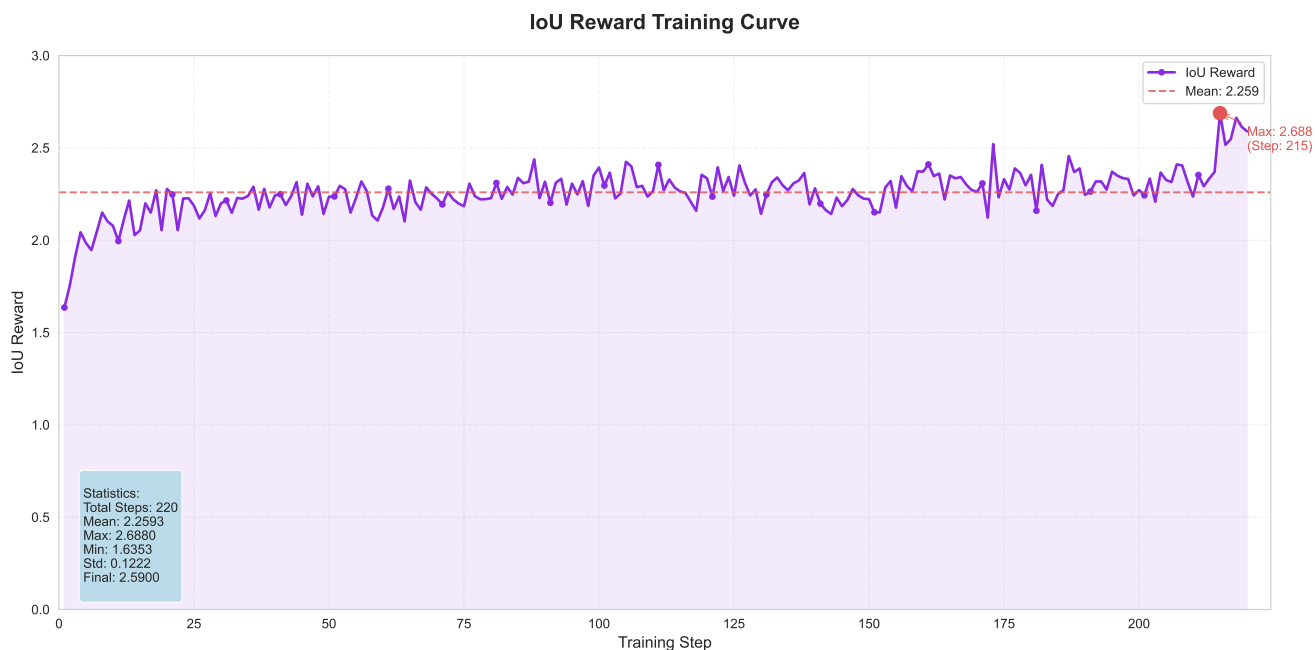


Figure 13. **The IoU reward curve.** We analyze the training dynamics to demonstrate that the RL training of IBISAgent show stable and consistent improvements.

### SYSTEM PROMPT

You are a precise and expert medical segmentation agent. Your mission is to accurately segment a target object in a medical image through a series of interactive point placements. You will be given an image and an instruction. You must carefully analyze the image state.

- If there is **no mask**, it is an **initialization** step. Your goal is to place a Positive Point on a clear, representative part of the target object.
- If a **semi-transparent green mask** is present, it is a **refinement** step. Your goal is to improve its accuracy. Place a Positive Point on a region of the target that the mask has missed, or a Negative Point on an area the mask has incorrectly included.

Your response must strictly follow this structure: first, your detailed reasoning within **<think>** tags, and then your single, decisive move within **<action>** tags. The only valid actions are **Positive Point (x, y)**, **Negative Point (x, y)**, or **Terminate**. All coordinates (x, y) must be normalized to a 0.0–1.0 scale and formatted to four decimal places, for example: Positive Point (0.5000, 0.2500). Only use Terminate when the mask perfectly aligns with the target boundary. If you Terminate, you must also append a final, concise summary in an **<answer>** tag.

### USER PROMPT

#### Initialization Prompts

- My goal is to segment the {obj}.
- Place the first point for {obj}.
- Where is the {obj} located?
- Where is the {obj} located in this image?
- Is it possible to get a segmentation of the {obj}?
- What would the segmentation for the {obj} look like?
- Visible {obj}? Please provide its segmentation.
- ...

#### Refinement Prompts

- Can you help me fix this segmentation of the {obj}?
- Let's keep working on this {obj}.
- Thanks. Now, what's the next refinement for the {obj}?
- The {obj} mask looks good, but can we improve it?
- Almost there with the {obj}. What's the final touch?
- The segmentation of the {obj} is ongoing. Please provide the next step.
- ...

Figure 14. The system and user prompt used in IBISAgent.

Table 11. Detailed statistics for the 38 task groups within our SFT dataset  $\mathcal{D}_{cold}$ .

Dataset (Group)	Samples	Avg. Length	Total QAs	Avg. IoU	Median IoU	Avg. DSC	Median DSC
ACDC	1,746	8.60	16,769	0.9453	0.9459	0.9718	0.9722
BreastUS	596	9.17	6,062	0.9351	0.9347	0.9664	0.9662
CAMUS	1,996	9.66	21,278	0.9405	0.9388	0.9693	0.9684
COVID-19_CT	224	9.08	2,258	0.8863	0.8826	0.9394	0.9376
COVID-QU_Ex	1,854	8.82	18,207	0.9584	0.9586	0.9787	0.9789
CXR_Masks_and_Labels	1,509	8.08	13,697	0.9470	0.9504	0.9727	0.9746
FH-PS-AOP	878	9.63	9,336	0.9203	0.9180	0.9584	0.9572
G1020	265	9.26	2,718	0.8834	0.8767	0.9377	0.9343
GlaS	123	8.65	1,187	0.7368	0.7336	0.8422	0.8463
ISIC	1,075	9.29	11,063	0.9331	0.9315	0.9653	0.9645
LGG	910	9.05	9,148	0.9341	0.9345	0.9658	0.9662
LIDC-IDRI	791	7.11	6,416	0.9254	0.9230	0.9612	0.9599
LiverUS	29	9.21	296	0.8940	0.9188	0.9428	0.9577
MMs	1,743	8.88	17,220	0.9484	0.9479	0.9735	0.9732
NeoPolyp	1,473	5.96	10,252	0.9561	0.9611	0.9775	0.9802
OCT-CME	205	8.02	1,850	0.9116	0.9074	0.9537	0.9515
PanNuke	1,072	8.87	10,578	0.9278	0.9267	0.9625	0.9620
PolypGen	958	6.50	7,182	0.9529	0.9603	0.9758	0.9797
QaTa-COV19	314	9.22	3,208	0.9049	0.8996	0.9500	0.9472
REFUGE	207	9.30	2,132	0.8833	0.8776	0.9379	0.9348
Radiography	4,156	8.93	41,254	0.9480	0.9524	0.9732	0.9756
UWaterlooSkinCancer	227	7.91	2,023	0.9430	0.9455	0.9706	0.9720
amos22	3,425	4.86	20,055	0.9640	0.9686	0.9816	0.9840
MSD brain tumor	1,925	5.94	13,366	0.9499	0.9490	0.9743	0.9738
MSD colon tumor	209	8.84	2,057	0.9138	0.9116	0.9549	0.9538
MSD heart	812	15.59	13,473	0.9095	0.9144	0.9524	0.9553
MSD hepatic vessel	306	18.47	5,959	0.8453	0.8405	0.9158	0.9133
MSD hepatic vessel tumor	53	9.34	548	0.7469	0.7495	0.8543	0.8568
MSD hippocampus	319	19.91	6,669	0.9358	0.9231	0.9665	0.9600
kits23	1,427	1.91	4,153	0.9712	0.9698	0.9854	0.9847
MSD liver	10,034	6.95	79,816	0.9549	0.9532	0.9769	0.9760
MSD liver tumor	546	8.07	4,951	0.9313	0.9287	0.9643	0.9630
MSD lung tumor	299	6.90	2,361	0.9352	0.9347	0.9664	0.9663
MSD pancreas	3,586	17.70	67,072	0.9240	0.9234	0.9604	0.9602
MSD pancreas tumor	474	7.50	4,030	0.9320	0.9314	0.9647	0.9645
MSD prostate	204	19.16	4,113	0.8664	0.8601	0.9279	0.9248
siim-acr-pneumothorax	252	9.18	2,565	0.8658	0.8593	0.9278	0.9243
MSD spleen	924	11.42	11,473	0.9458	0.9507	0.9721	0.9747
<b>Total</b>	<b>47,146</b>	<b>8.69</b>	<b>456,795</b>	<b>0.9427</b>	<b>0.9507</b>	<b>0.9703</b>	<b>0.9747</b>

