This MICCAI paper is the Open Access version, provided by the MICCAI Society. It is identical to the accepted version, except for the format and this watermark; the final published version is available on SpringerLink.

Learning to Reason Over Physician Corrections: An Interactive Agentic Framework for 3D Tumor Segmentation

Nourhan Bayasi[1,2][0000−0003−4653−6081], Fereshteh Yousefirizi[1,2][0009−0002−3461−2606], and Arman Rahmim[1,2][0000−0002−9980−2403]

[1] University of British Columbia, Vancouver, BC, Canada
[2] BC Cancer Research Institute, Vancouver, BC, Canada
{nbayasi,frizi,arahmim}@bccrc.ca

Abstract. Interactive segmentation models ask a physician to iteratively correct a model's output until the result is acceptable, yet existing methods treat each correction as an isolated reactive event with no structured reasoning about the interaction itself. In this paper, we propose LORE, an interactive agentic framework built on a frozen 3D SAM backbone that trains an agent to reason over the full history of physician corrections so that it can ask for the right kind of correction at the right time. The agent learns through the Chain of Clinical Corrections (CoCC): it observes the geometry of the current error, accumulates correction patterns across the session to detect recurring structural failures, hypothesizes where the next correction will fall, and verifies its own spatial prediction against the physician's subsequent action via a novel spatial grounding reward, making the agent rewarded not only for segmentation quality, but for correctly anticipating where the physician will need to act next. Evaluated across six benchmarks spanning CT (colon, pancreas, liver, kidney) and FDG-PET (head-and-neck, lung), LORE consistently achieves strong segmentation quality while substantially reducing physician interaction burden. Code is available here.

Keywords: agentic AI · physician-in-the-loop · medical image segmentation · segment anything model · clinical workflow.

1 Introduction

Unlike automated medical image segmentation models [12], interactive models including SAM-based frameworks, e.g., SAM-Med3D [27] and MedSAM [23], allow a physician to iteratively correct a model's output via clicks, boxes, or scribbles until the result is acceptable [9]. These methods have demonstrated strong performance across diverse anatomies and modalities, yet they share a critical limitation: each correction is processed independently, with no structured reasoning of what went wrong or about the interaction itself. These methods cannot distinguish a one-off localized miss from a recurring structural failure, have no way to anticipate where the next correction will be needed, and apply the

same passive update regardless of how many times they have failed in the same region. In busy clinical workflows, where each correction costs physician time and attention, a constraint that spans clinical domains from imaging review to continuous patient monitoring [10], this stateless design demands more interactions than necessary, and the more interactions it forces, the more it degrades the precision of the feedback it receives.

Recent reinforcement learning (RL)-based methods take a step forward by learning a policy that selects which prompt type to request at each step. AIES [16] formulates interactive segmentation as an MDP, choosing among box, click, and stop actions based on Dice change and interaction cost. AlignSAM [15] learns to select point locations from image-embedding state. Both reduce unnecessary interactions, but share a fundamental constraint: their policy state captures only a snapshot of the current step and its segmentation outcome, with no memory of how errors evolve across the session, no anticipation of where the next correction will fall, and no reward for intermediate reasoning quality. The policy therefore has no basis for directing the physician toward the most likely error location, or for recognising when the model is caught in a recurring failure pattern.

To overcome the aforementioned limitations, we propose `LORE`, an interactive agentic framework built on a frozen 3D SAM backbone that trains an RL agent to reason over physician corrections in order to understand how errors evolve and how the physician responds to them so that at inference, it asks for only the right prompt at the right time. The core idea is simple: the agent builds a structured understanding of the interaction session by observing what the model got wrong and in what shape, remembering whether the same failure has occurred before, predicting where the next correction is most likely to fall, and verifying its own prediction against what the physician actually does next. This four-step reasoning process, which we term the **Chain of Clinical Corrections (CoCC)**, makes `LORE` the first interactive segmentation framework in which the agent is held accountable not only for improving segmentation quality, but for deciding correctly when and how to involve the physician, closing a reasoning loop entirely absent from prior work. We validate `LORE` across six benchmarks spanning CT (kidney, liver, colon, pancreas) and FDG-PET (head-and-neck, lung), demonstrating consistent improvements in segmentation quality and interaction efficiency compared to baselines and recent state-of-the-art interactive models.

2 Methodology

`LORE` formalizes the interaction session as a reasoning trace through CoCC, instantiated as a frozen 3D SAM backbone paired with a learned CoCC reasoning pipeline and a learned RL agent. An overview is illustrated in Fig. 1.

2.1 Problem Formulation. Let $\mathcal{I} \in \mathbb{R}^{C\times D\times H\times W}$ denote a volumetric medical image and $\mathbf{y} \in \{0,1\}^{D\times H\times W}$ its binary segmentation mask. At each interaction step $i \in \{1,\ldots,T\}$, where T is the maximum number of allowed interactions, the physician provides guidance $\mathbf{g}_i$ (e.g., a click, bounding box, or scribble) and the backbone produces an updated segmentation $\hat{\mathbf{y}}_i = f_\theta(\mathcal{I}, \mathbf{g}_{0:i}, \hat{\mathbf{y}}_{i-1})$. Rather

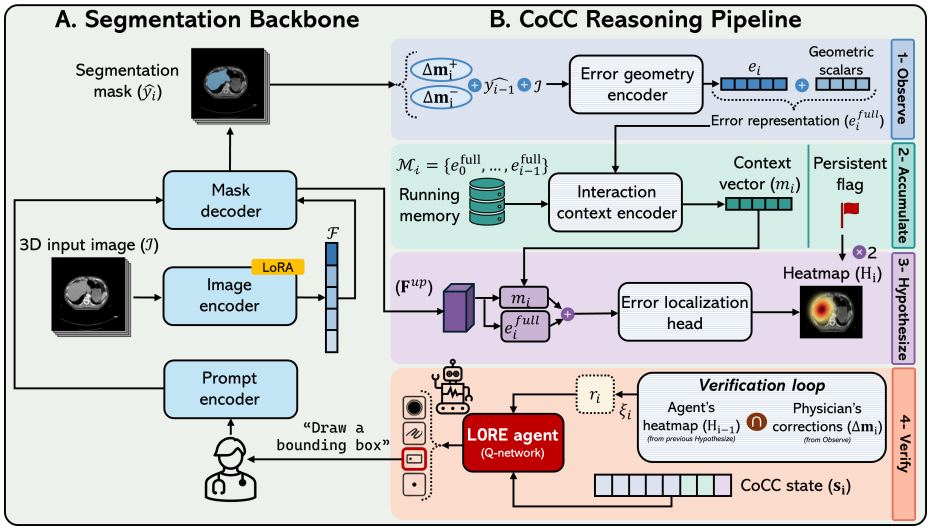

Fig. 1: **Overview of LORE.** (A) A frozen SAM with lightweight LoRA adapters produces the segmentation estimate $\hat{\mathbf{y}}_i$. (B) The CoCC reasoning pipeline constructs a structured reasoning state through four steps: Observe, Accumulate, Hypothesize, and Verify, which the LORE agent (Q-network) uses to select the optimal prompt type a_i, supervised by the spatial grounding reward $\xi_i \subset r_i$.

than optimizing segmentation accuracy under a fixed interaction budget, we model this as an adaptive process with two coupled objectives: (i) maximize segmentation quality, measured by the Dice Similarity Coefficient d_i; and (ii) minimize cumulative physician effort $\sum_i \ell(a_i)$, where $a_i \in \mathcal{A}$ is the interaction type selected at step i and $\ell(\cdot)$ is an effort cost function. During training, a hard quality floor masks stop whenever $d_i < \tau_{\text{stop}}$ (with d_i computed against $\mathbf{y}$), shaping the policy to avoid premature termination.

2.2 Segmentation Backbone. We adopt the 3D SAM architecture [21], keeping its parameters frozen throughout training. Lightweight LoRA adapters are inserted into the image encoder to enable parameter-efficient adaptation without modifying the pretrained feature representations. The image encoder produces a multi-scale image embedding $\mathbf{F}$. A prompt encoder supports point clicks, bounding boxes, and scribble masks, with all accumulated prompts $\mathbf{g}_{0:i}$ passed jointly at each step i. A bidirectional cross-attention mask decoder produces a segmentation mask $\hat{\mathbf{y}}_i$, which is passed to the CoCC pipeline.

2.3 CoCC Reasoning Pipeline.

2.3.1. Step 1: Observe. The Observe step answers: *what did the model get wrong at this step, and what does the shape of that mistake reveal?* The agent receives $\hat{\mathbf{y}}_i$ and forms false-negative ($\Delta\mathbf{m}_i^+$) and false-positive ($\Delta\mathbf{m}_i^-$) maps against the ground-truth mask $\mathbf{y}$. These maps, concatenated with the previous prediction and the input image, are passed through an error geometry encoder, yielding

a correction embedding $\mathbf{e}_i$. In addition, four interpretable geometric scalars are appended: correction volume v_i, asymmetry ρ_i (FN-to-FP ratio), compactness κ_i (error sphericity), and boundary overlap b_i (fraction of the error within one voxel of the prediction boundary), producing the full error representation $\mathbf{e}_i^{\text{full}} \in \mathbb{R}^{132}$.

2.3.2. Step 2: Accumulate. The Accumulate step answers: *is this a new failure or one the model has shown before?* Inspired from continual learning [8,3,7,4,5], the agent maintains a running memory $\mathcal{M}_i = \{\mathbf{e}_0^{\text{full}}, \ldots, \mathbf{e}_{i-1}^{\text{full}}\}$ of all past correction embeddings, with positional encodings indexed by interaction number. An interaction context encoder aggregates this sequence into a session-level context vector $\mathbf{m}_i$ by attending over all past corrections, naturally surfacing which error patterns recur. After a minimum number of steps, the agent computes pairwise cosine similarity between the current embedding ($\mathbf{e}_i^{\text{full}}$) and each prior embedding ($\mathbf{e}_j^{\text{full}}$, $j < i$); if any similarity exceeds τ_{persist}, a persistence flag is raised, identifying the current failure as a structural blind spot [6].

2.3.3. Step 3: Hypothesize. The Hypothesize step answers: *where is the next correction most likely to be needed?* The agent uses an error localization head to predict a 3D heatmap $\mathbf{H}_i \in [0,1]^{D\times H\times W}$ over the volume before the physician begins searching. The head operates on the mask decoder's upsampled features $\mathbf{F}^{up}$, conditioned on $\mathbf{e}_i^{\text{full}}$ and $\mathbf{m}_i$ via Feature-wise Linear Modulation (FiLM):

$$\mathbf{x}_1 = (1+\gamma^e)\odot\mathbf{F}^{up}+\beta^e, \quad \mathbf{x}_2 = (1+\gamma^m)\odot\phi_1(\mathbf{x}_1)+\beta^m, \quad \mathbf{H}_i = \sigma(\text{Conv}(\phi_2(\mathbf{x}_2)))\,, \tag{1}$$

where σ denotes the sigmoid, (γ^e, β^e) and (γ^m, β^m) are affine parameters projected from stop-gradient copies of $\mathbf{e}_i^{\text{full}}$ and $\mathbf{m}_i$ respectively, and ϕ_1, ϕ_2 are residual convolutional blocks. The stop-gradient prevents RL gradients from propagating into the backbone's spatial representations. The heatmap is supervised against a Gaussian target centered on the centroid of the next-step error region; when the persistence flag f_{persist} is raised, the heatmap is additionally scaled by a factor of two, boosting predicted uncertainty in the region the model is actively failing on. Its normalized Shannon entropy u_i enters the CoCC state as the spatial uncertainty scalar, and its overlap with the physician's actual next correction forms the spatial grounding reward ξ_i, closing the Verify loop.

2.3.4. Step 4: Verify. The Verify step answers: *did the agent ask for the right correction at the right time, and did it correctly anticipate where the physician would act?* The agent assembles the outputs of Steps 1–3 into a compact reasoning state, selects the next action, and retrospectively scores its own spatial prediction $\mathbf{H}_{i-1}$ against the physician's actual correction at step i via the spatial grounding reward ξ_i.

A. CoCC state. The outputs of the error geometry encoder ($\mathbf{e}_i^{\text{full}}$), the interaction context encoder ($\mathbf{m}_i$), and the error localization head ($\mathbf{H}_{i-1}$) are distilled into an 8-scalar state: $\mathbf{s}_i = (d_i, \Delta d_i, i/T, v_i, b_i, \|\mathbf{e}_i^{\text{full}}\|/(\|\mathbf{e}_i^{\text{full}}\|+1), \cos(\mathbf{e}_i^{\text{full}}, \mathbf{m}_i), u_{i-1})$, encoding Dice, Dice gain, budget usage, error volume, boundary overlap, error severity, persistence, and spatial uncertainty.

B. Policy and action space. A compact MLP maps $\mathbf{s}_i$ to Q-values over $\mathcal{A} = \{\text{point, box, scribble, stop}\}$, with a target network providing stable Bellman targets under Double DQN. Two hard constraints are enforced during training

and simulated evaluation: scribble is suppressed for large-volume errors; stop is masked when $d_i < \tau_{\text{stop}}$. The r_{stop} term (paragraph C) remains defined below τ_{stop} only to regularize masked-regime Q-values for Double DQN; that branch is never reached in trajectory collection.

C. Reward. The reward balances five clinical objectives at each step i:

$$r_i = \underbrace{\Delta d_i}_{\text{Dice gain}} - \underbrace{\ell(a_i)(1+v_i)}_{\text{physician burden}} + \underbrace{\alpha\, b_i \mathbb{1}[a_i = \text{scribble}]}_{\text{boundary targeting}} + \underbrace{r_{\text{stop}}(a_i, d_i)}_{\text{termination quality}} + \underbrace{\beta\, \xi_i \mathbb{1}[a_i \neq \text{stop}]}_{\text{spatial grounding}}. \tag{2}$$

The agent is rewarded for improving segmentation (Δd_i), penalized for costly prompt types scaled by remaining error volume ($\ell(a_i)(1+v_i)$), and additionally rewarded for targeting boundary errors with scribbles (b_i) or stopping at the right quality: $r_{\text{stop}}(a_i, d_i)$=+0.5 when a_i=stop and $d_i \geq \tau_{\text{stop}}$, −0.5 when a_i=stop and $d_i < \tau_{\text{stop}}$, and 0 otherwise. The negative branch is unreachable in practice (stop is masked below τ_{stop}, as noted above) but stabilizes Double DQN Bellman targets by keeping the target network's Q-value for stop low in that regime. The fifth term ξ_i closes the Verify loop by measuring whether the heatmap $\mathbf{H}_{i-1}$ correctly anticipated where the physician would correct at step i:

$$\xi_i = \frac{\sum_x \mathbf{H}_{i-1}(x) \cdot \mathbb{1}\left[x \in \Delta\mathbf{m}_i^+ \cup \Delta\mathbf{m}_i^-\right]}{\left|\Delta\mathbf{m}_i^+ \cup \Delta\mathbf{m}_i^-\right| + \epsilon}. \tag{3}$$

When $\xi_i \approx 1$, the agent predicted correctly; when $\xi_i \approx 0$, it did not. Unlike all prior RL-based interactive segmentation methods, which reward only Δd_i, the ξ_i term holds the agent accountable for the quality of its intermediate reasoning, not just the quality of the final output.

2.4 Training & Inference. The full objective is:

$$\mathcal{L} = \mathcal{L}_{\text{seg}} + \mathcal{L}_{\text{bnd}} + \lambda_s(t)\, \mathcal{L}_{\text{spatial}} + \lambda_{\text{rl}}(t)\, \mathcal{L}_{\text{DQN}}, \tag{4}$$

where $\lambda_s(t)$ and $\lambda_{\text{rl}}(t)$ are curriculum-scheduled weights that ramp up from zero over training. $\mathcal{L}_{\text{seg}}$ is a DiceCE loss (Dice + binary cross-entropy) between the predicted mask and ground truth; $\mathcal{L}_{\text{bnd}}$ is a boundary sharpening loss computed as the MSE between the predicted and ground-truth boundary maps obtained by subtracting a locally-pooled version of each mask from itself; $\mathcal{L}_{\text{spatial}}$ is the MSE between the predicted heatmap $\mathbf{H}_i$ and the Gaussian target centered on the next-step correction centroid, supervising the error localization head; and $\mathcal{L}_{\text{DQN}}$ [26] trains the prompt selection policy via experience replay to minimize the Huber loss between the current Q-estimate and the Double DQN target [26,24]: $\mathcal{L}_{\text{DQN}} = \mathbb{E}_{(s,a,r,s')\sim\mathcal{B}}[\ell_\delta(r + \gamma\, Q_{\bar{\theta}}(s', \arg\max_{a'} Q_\theta(s', a')) - Q_\theta(s, a))]$, where Q_θ is the online network and $Q_{\bar{\theta}}$ the target network. The backbone optimizer trains the LoRA layers and error localization head (Step 3) via $\mathcal{L}_{\text{seg}}+\mathcal{L}_{\text{bnd}}+\lambda_s(t)\,\mathcal{L}_{\text{spatial}}$; the policy optimizer trains the error geometry encoder (Step 1), interaction context encoder (Step 2), and Q-network (Step 4) via $\mathcal{L}_{\text{DQN}}$. Following standard interactive-segmentation practice [21,16], all GT-dependent signals (Observe FN/FP maps, d_i state features, τ_{stop} masking, and r_{stop}) are used only during training and simulated evaluation. At clinical inference, after an initial click

produces $\hat{\mathbf{y}}_0$, LORE runs CoCC without GT: Observe uses the localized discrepancy between $\hat{\mathbf{y}}_i$ and the physician's correction $\mathbf{g}_i$, and $a_i = \arg\max_a Q(\mathbf{s}_i, a)$ (ε=0) with stopping driven by the learned Q-policy (τ_{stop} internalized in training); the physician provides the next requested correction or confirms the result.

3 Experiments and Results

3.1 Datasets. We use four CT tumor segmentation datasets: **Colon** (126 volumes) and **Pancreas** (281 volumes) from the Medical Segmentation Decathlon (MSD) [2], **Liver** (118 volumes) from LiTS [11], and **Kidney** (268 volumes) from KiTS21 [14]; and two FDG-PET datasets: **Head-and-Neck** (400 volumes) [1] provides gross tumor volume segmentation from the HECKTOR 2022 challenge, and **Lung** (169 volumes) [19] drawn from the AutoPET dataset. All CT volumes are resampled to $1\,\text{mm}^3$ isotropic spacing, clipped and z-score normalized per dataset; PET volumes are retained at native spacing with percentile-based intensity scaling. All inputs are processed as 128^3 patches. We follow prior work [21] and apply a 0.7/0.1/0.2 train/val/test split per dataset.

3.2 Implementation Details. The backbone uses a frozen 3D ViT-B encoder (r=4 LoRA adapters) with hybrid U-Net skip connections. Step 1 uses a 3D ResNet stem (4→32→64→128, GAP) projecting to $\mathbf{e}_i^{\text{full}} \in \mathbb{R}^{132}$. Step 2 uses a 2-layer Transformer (dim 132, 4 heads, FFN 256, sinusoidal PE); persistence flag at τ_{persist}=0.85 after $i \geq 3$ steps. Step 3 applies two residual convolutional blocks supervised by a Gaussian target (std=5 voxels). Step 4 is a 3-layer MLP (8→32→16→4). Reward: α=0.3, β=0.25, τ_{stop}=0.70. DQN: buffer 10k, batch 64, γ=0.99, τ=0.005. Two AdamW optimizers train separately: backbone ($\eta = 4\times10^{-5}$, linear decay; updates LoRA adapters and localization head) and policy ($\eta = 10^{-4}$; updates Steps 1–2 encoders and Q-network), with staged curricula for noise annealing, spatial loss ramp, and RL activation.

3.3 Baselines, Competing Methods & Evaluation Metrics. We compare against: fully automated (**nnU-Net** [17], **3D UX-Net** [20], **SwinUNETR** [25]); expert-based interactive (**SAM** [18], **PRISM** [21], **3DSAM-adapter** [13], **ProMISe** [22], **SAM-Med3D** [27]), each with up to 11 simulated prompts following prior works [21]; and RL-based interactive methods (**AIES** [16]). We report Dice (d), normalized surface Dice (NSD, 5 mm), mean interactions ($\bar{K}$), Clinical Efficiency Score ($CES = d/(\bar{K}+1)$), and weighted effort $\bar{E}$ ($\ell \in \{0.1, 0.3, 0.5\}$ for point, box, scribble).

3.4 Main Results. From Table 1, LORE achieves the best Dice on all six datasets and the best NSD on five of six, with the lowest $\bar{K}$ on four of six. Against AIES, the largest gains are on H&N (+4.7 pp Dice, +5.6 pp NSD) and Pancreas (+3.4 pp Dice, +5.6 pp NSD); on Liver, LORE nearly halves interactions ($\bar{K}$=4.5 vs. 8.8) while still improving Dice (+3.0 pp). Relative to expert methods under a fixed 11-step budget, LORE raises Dice by up to 17 pp (Pancreas vs. PRISM; Liver vs. SAM-Med3D-turbo) at under half the interaction cost. Where AIES stops earlier on Pancreas and Lung ($\bar{K}$=5.1 and 5.5 vs. 7.1 and 8.0), it does so at substantially lower Dice, a trade-off CES is designed to penalize.

Table 1: Segmentation results across six benchmarks reporting Dice (d), normalized surface distance (NSD), and mean interactions ($\bar{K}$). Fully automated methods require no physician interaction (—). Best in **bold**, second best underlined.

	d % (↑) / NSD % (↑) / $\bar{K}$ (↓)					
Method	**Colon**	**Pancreas**	**Liver**	**Kidney**	**H&N**	**Lung**
Fully automated						
nnU-Net [17]	43.91 / 52.52 / —	41.65 / 62.54 / —	60.10 / 75.41 / —	73.07 / 77.47 / —	57.34 / 68.92 / —	51.27 / 62.43 / —
3D UX-Net [20]	28.50 / 32.73 / —	34.83 / 52.56 / —	45.54 / 60.67 / —	57.59 / 58.55 / —	44.21 / 54.37 / —	38.94 / 47.82 / —
SwinUNETR [25]	35.21 / 42.94 / —	40.57 / 60.05 / —	50.26 / 64.33 / —	65.54 / 72.04 / —	51.83 / 63.74 / —	46.73 / 58.31 / —
Expert-based prompting methods						
SAM [18]	28.83 / 33.63 / 11.0	24.01 / 26.74 / 11.0	8.56 / 5.97 / 11.0	36.30 / 29.86 / 11.0	31.47 / 35.62 / 11.0	24.83 / 28.47 / 11.0
3DSAM-adapter [13]	57.32 / 73.65 / 11.0	54.41 / 77.88 / 11.0	56.61 / 69.52 / 11.0	73.78 / 83.86 / 11.0	62.18 / 74.53 / 11.0	55.34 / 67.82 / 11.0
ProMISe [22]	66.81 / 81.24 / 11.0	57.46 / 79.76 / 11.0	58.78 / 71.52 / 11.0	75.70 / 80.08 / 11.0	68.42 / 80.17 / 11.0	61.47 / 73.25 / 11.0
SAM-Med3D [27]	54.34 / 78.58 / 11.0	65.61 / 92.40 / 11.0	23.64 / 26.97 / 11.0	76.50 / 88.41 / 11.0	55.63 / 71.84 / 11.0	48.92 / 63.41 / 11.0
SAM-Med3D-organ	70.75 / 91.03 / 11.0	76.40 / 97.75 / 11.0	66.52 / 77.97 / 11.0	88.20 / 97.80 / 11.0	73.21 / 86.45 / 11.0	65.83 / 78.42 / 11.0
SAM-Med3D-turbo	73.77 / **94.95** / 11.0	74.87 / 96.43 / 11.0	69.36 / 81.70 / 11.0	89.26 / 98.30 / 11.0	79.47 / 89.23 / 11.0	67.24 / 81.35 / 11.0
PRISM [21]	67.18 / 85.28 / 11.0	65.73 / 89.51 / 11.0	79.70 / 91.60 / 11.0	85.29 / 93.55 / 11.0	71.83 / 84.62 / 11.0	64.38 / 76.84 / 11.0
Automated prompting						
AIES [16]	75.42 / 89.63 / 5.4	79.18 / 93.45 / **5.1**	83.27 / 92.14 / 8.8	91.34 / 96.82 / 4.3	76.83 / 88.41 / 5.7	67.25 / 79.34 / **5.5**
LORE **(Ours)**	**76.97** / 93.46 / **4.3**	**82.56** / **99.04** / 7.1	**86.29** / **93.62** / **4.5**	**92.93** / **98.33** / **3.2**	**81.53** / **93.98** / **4.8**	**69.31** / **85.01** / 8.0

Table 2: Interaction efficiency and prompt-type comparison. *Left block*: agent efficiency metrics ($\bar{K}$, $\bar{E}$, CES, action distribution). *Right block*: Δd = agent Dice minus forced-action Dice (pp); positive ↑ means agent wins quality.

	Agent efficiency				*vs. Fixed Box*		*vs. Fixed Scribble*	
Dataset	$\bar{K}$↓	$\bar{E}$↓	CES↑	Pt/Bx/Sc %	Δd (pp)	$\bar{E}$	Δd (pp)	$\bar{E}$
Colon	4.3	0.46	0.145	96/4/0	−0.1	1.20	+2.1↑	2.00
Pancreas	7.1	0.71	0.102	100/0/0	+0.3↑	2.10	−0.7	3.50
Liver	4.5	0.95	0.157	72/0/28	+18.0↑	1.50	+21.7↑	2.50
Kidney	3.2	0.32	0.221	100/0/0	+10.0↑	0.90	+10.7↑	1.50
H&N	4.8	0.77	0.141	85/0/15	+42.3↑	1.50	+34.0↑	2.50
Lung	8.0	2.22	0.077	11/89/0	+6.6↑	0.80	+7.7↑	4.00

Table 3: Ablation study (mean over six benchmarks). Architecture group ablates CoCC components; reward group ablates training terms.

Variant	d (↑)	$\bar{K}$ (↓)	CES (↑)
Architecture ablations			
LORE **(Full)**	**81.60**	**5.32**	**0.141**
w/o adaptive stopping (fixed $\bar{K}$=11)	78.15	11.00	0.064
w/o Accumulate	77.86	6.88	0.098
w/o Hypothesize	73.24	7.82	0.082
w/o Geometric scalars	79.96	5.71	0.119
Reward ablations			
w/o ξ_i (spatial grounding)	78.79	6.19	0.110
w/o $\ell(a_i)$ (physician burden)	81.22	8.53	0.085
w/o r_{stop} (termination quality)	78.04	7.09	0.096

3.4.1 Interaction Efficiency. Table 2 (left) details LORE's efficiency profile. Kidney achieves the lowest effort ($\bar{E}$=0.32, $\bar{K}$=3.2) and highest CES (0.221), confirming early, accurate termination on well-defined renal tumors. Liver achieves high Dice (86.29%) at moderate cost ($\bar{E}$=0.95, CES=0.157) with a mixed point-scribble strategy suited to irregular lesion boundaries. Pancreas relies exclusively on point prompts (100%) and reaches near-optimal quality in $\bar{K}$=7.1 steps ($\bar{E}$=0.71). Lung incurs the highest effort ($\bar{E}$=2.22) with a box-dominant strategy (89%) reflecting diffuse, low-contrast FDG-PET uptake regions.

3.4.2 Prompt-Type Comparison. Table 2 (right) benchmarks the adaptive policy against fixed-action baselines at the same $\bar{K}$ budget. The Δd column shows that the agent matches or exceeds fixed alternatives in Dice on five of six datasets; the single exception (Colon vs. box, Δd=−0.1 pp) is negligible. Agent effort ($\bar{E}$, teal column) is consistently lower than forced-box and forced-scribble alternatives, with savings ranging from 1.6× (Liver) up to 5× (Pancreas): forced scribble on Pancreas achieves similar Dice but at 5× the effort ($\bar{E}$=3.50 vs. 0.71); forced box on Kidney costs 2.8× more ($\bar{E}$=0.90 vs. 0.32) while losing 10 pp Dice. Lung is the instructive exception: the agent's box-dominant strategy uses more effort than forced-point ($\bar{E}$=2.22 vs. 0.80) but gains 6.6 pp Dice, showing that

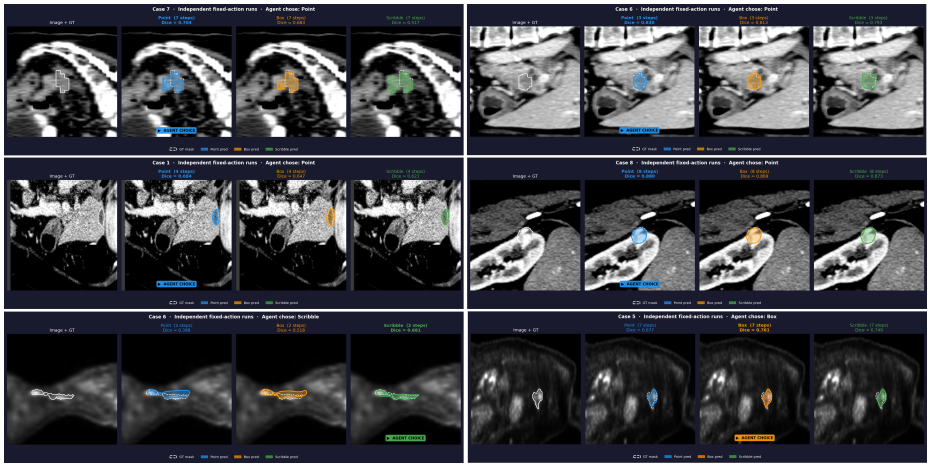

Fig. 2: Qualitative prompt-type comparison (one representative case per dataset). Each row has four panels: (1) CT/PET slice with ground-truth contour; (2)–(4) segmentations under forced point, box, and scribble prompts at the same interaction budget $\bar{K}$, with Dice shown above each panel. The *Agent Choice* badge marks the prompt type selected by `LORE`.

the policy trades effort for quality only when the anatomy demands it. Fig. 2 provides qualitative confirmation: the agent's chosen prompt type consistently produces tighter, more complete segmentations than the fixed alternatives at the same interaction budget.

3.5 Ablation Studies. Table 3 reports mean d, $\bar{K}$, and CES over six benchmarks, ablating CoCC components and individual reward terms.

Architecture ablations. Disabling adaptive stopping (fixed $\bar{K}$=11) inflates $\bar{K}$ by +5.7 and drops CES to 0.064, showing learned termination drives efficiency. Without Accumulate (-3.7 pp d, $+1.6$ $\bar{K}$), the agent cannot detect recurring structural failures. Without Hypothesize (no heatmap, no ξ_i), d falls most (-8.4 pp) and $\bar{K}$ rises +2.5, confirming spatial grounding as the main quality driver. Removing geometric scalars has the mildest effect (-1.6 pp d), as CNN-encoded $\mathbf{e}_i$ partially substitutes.

Reward ablations. Removing ξ_i from the reward while keeping the heatmap in state drops d by 2.8 pp, isolating spatial accountability from uncertainty as a policy cue. Removing $\ell(a_i)$ barely affects d (-0.4 pp) but raises $\bar{K}$ most (+3.2; lowest CES=0.085), as the policy over-uses costly prompts. Removing r_{stop} degrades both d (-3.6 pp) and $\bar{K}$ (+1.8), impairing termination judgment.

4 Conclusion

We propose `LORE`, an interactive agentic framework that replaces the passive interaction loop of physician-in-the-loop segmentation with structured clinical

reasoning. Through the Chain of Clinical Corrections, LORE learns to anticipate where the physician will correct next and which prompt type to request, closing a reasoning loop entirely absent from prior work. Across six CT and FDG-PET tumor benchmarks, LORE achieves the best trade-off between segmentation quality and interaction efficiency, demonstrating that reasoning over correction history yields stronger segmentation with a more efficient clinical workflow.

References

1. Andrearczyk, V., Oreiller, V., Abobakr, M., Akhavanallaf, A., Balermpas, P., Boughdad, S., Capriotti, L., Castelli, J., Cheze Le Rest, C., Decazes, P., et al.: Overview of the hecktor challenge at miccai 2022: automatic head and neck tumor segmentation and outcome prediction in pet/ct. In: 3D Head and Neck Tumor Segmentation in PET/CT Challenge, pp. 1–30 (2022)
2. Antonelli, M., Reinke, A., Bakas, S., Farahani, K., Kopp-Schneider, A., Landman, B.A., Litjens, G., Menze, B., Ronneberger, O., Summers, R.M., et al.: The medical segmentation decathlon. Nature communications **13**(1), 4128 (2022)
3. Bayasi, N., Du, S., Hamarneh, G., Garbi, R.: Continual-gen: Continual group ensembling for domain-agnostic skin lesion classification. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 3–13. Springer (2023)
4. Bayasi, N., Fayyad, J., Bissoto, A., Hamarneh, G., Garbi, R.: Biaspruner: Debiased continual learning for medical image classification. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 90–101. Springer (2024)
5. Bayasi, N., Hamarneh, G., Garbi, R.: Culprit-prune-net: Efficient continual sequential multi-domain learning with application to skin lesion classification. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 165–175. Springer (2021)
6. Bayasi, N., Hamarneh, G., Garbi, R.: Boosternet: Improving domain generalization of deep neural nets using culpability-ranked features. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 538–548 (2022)
7. Bayasi, N., Hamarneh, G., Garbi, R.: Continual-zoo: Leveraging zoo models for continual classification of medical images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4128–4138 (2024)
8. Bayasi, N., Hamarneh, G., Garbi, R.: Gc 2: Generalizable continual classification of medical images. IEEE Transactions on Medical Imaging **43**(11), 3767–3779 (2024)
9. Bayasi, N., Pouromidi, M., Yousefirizi, F., Rahmim, A.: Interactive personalized ai for physician-in-the-loop 3d tumor segmentation on ct. In: 2026 Joint AAPM| COMP Annual Meeting & Exhibition. AAPM (2026)
10. Bayasi, N., Saleh, H., Mohammad, B., Ismail, M.: The revolution of glucose monitoring methods and systems: A survey. In: 2013 IEEE 20th International Conference on Electronics, Circuits, and Systems (ICECS). pp. 92–93 (2013)
11. Bilic, P., Christ, P., Li, H.B., Vorontsov, E., Ben-Cohen, A., Kaissis, G., Szeskin, A., Jacobs, C., Mamani, G.E.H., Chartrand, G., et al.: The liver tumor segmentation benchmark (lits). Medical image analysis **84**, 102680 (2023)
12. Du, S., Bayasi, N., Hamarneh, G., Garbi, R.: Avit: Adapting vision transformers for small skin lesion segmentation datasets. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 25–36 (2023)

13. Gong, S., Zhong, Y., Ma, W., Li, J., Wang, Z., Zhang, J., Heng, P.A., Dou, Q.: 3dsam-adapter: Holistic adaptation of sam from 2d to 3d for promptable tumor segmentation. Medical Image Analysis **98**, 103324 (2024)
14. Heller, N., Isensee, F., Maier-Hein, K.H., Hou, X., Xie, C., Li, F., Nan, Y., Mu, G., Lin, Z., Han, M., et al.: The state of the art in kidney and kidney tumor segmentation in contrast-enhanced ct imaging: Results of the kits19 challenge. Medical image analysis **67**, 101821 (2021)
15. Huang, D., Xiong, X., Ma, J., Li, J., Jie, Z., Ma, L., Li, G.: Alignsam: Aligning segment anything model to open context via reinforcement learning. In: IEEE CVPR
16. Huang, Y., Shen, C., Li, W., Wang, X., Jin, B., Cai, H.: Optimizing efficiency and effectiveness in sequential prompt strategy for sam using reinforcement learning. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 478–488 (2024)
17. Isensee, F., Jaeger, P.F., Kohl, S.A., Petersen, J., Maier-Hein, K.H.: nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. Nature methods **18**(2), 203–211 (2021)
18. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. In: IEEE CVPR
19. Küstner, T., Gatidis, S., Megne, O., Ingrisch, M., Fabritius, M., Dexl, J., Jeblick, K., Mittermeier, A., Schachtner, B., Stüber, T., Cyran, C., Marinov, Z., Peisen, F., Wagner, A., Othman, A., Sanner, A., Loßau, T., Moltz, J.H., Kohlbrandt, T., Hering, A.: Automated lesion segmentation in whole-body pet/ct and longitudinal (autopet/ct iv) (2025)
20. Lee, H.H., Bao, S., Huo, Y., Landman, B.A.: 3d ux-net: A large kernel volumetric convnet modernizing hierarchical transformer for medical image segmentation. arXiv preprint arXiv:2209.15076 (2022)
21. Li, H., Liu, H., Hu, D., Wang, J., Oguz, I.: Prism: A promptable and robust interactive segmentation model with visual prompts. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 389–399 (2024)
22. Li, H., Liu, H., Hu, D., Wang, J., Oguz, I.: Promise: Prompt-driven 3d medical image segmentation using pretrained image foundation models. In: IEEE international symposium on biomedical imaging (ISBI). pp. 1–5 (2024)
23. Ma, J., He, Y., Li, F., Han, L., You, C., Wang, B.: Segment anything in medical images. Nature Communications **15**(1), 654 (2024)
24. Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., et al.: Human-level control through deep reinforcement learning. nature **518**(7540), 529–533 (2015)
25. Tang, Y., Yang, D., Li, W., Roth, H.R., Landman, B., Xu, D., Nath, V., Hatamizadeh, A.: Self-supervised pre-training of swin transformers for 3d medical image analysis. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 20730–20740 (2022)
26. Van Hasselt, H., Guez, A., Silver, D.: Deep reinforcement learning with double q-learning. In: Proceedings of the AAAI conference on artificial intelligence. vol. 30 (2016)
27. Wang, H., Guo, S., Ye, J., Deng, Z., Cheng, J., Li, T., Chen, J., Su, Y., Huang, Z., Shen, Y., et al.: Sam-med3d: a vision foundation model for general-purpose segmentation on volumetric medical images. IEEE Transactions on Neural Networks and Learning Systems (2025)