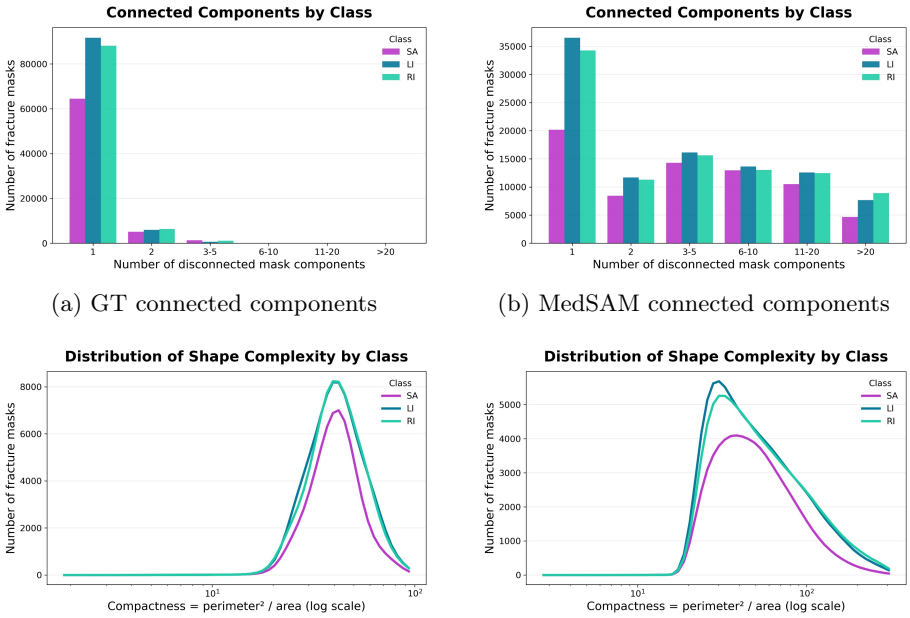
Appendix

A Exploratory Data Analysis



(a) GT connected components

(b) MedSAM connected components

(c) GT compactness

(d) MedSAM compactness

Fig. 6. Comparison of shape statistics between ground-truth fracture masks and MedSAM predictions. Top row: distribution of connected components per fragment. Bottom row: distribution of compactness values. Compared to the ground-truth masks, MedSAM predictions exhibit substantially increased fragmentation and greater boundary irregularity, reflected by broader connected-component and compactness distributions. These geometric instabilities are especially pronounced for small fracture fragments.

Figure 6 compares geometric shape statistics between ground-truth fracture masks and MedSAM predictions. While ground-truth fragments are predominantly connected and geometrically compact, MedSAM predictions exhibit substantially increased fragmentation and boundary irregularity. In particular, MedSAM masks contain more disconnected components and broader compactness distributions, indicating noisier and less stable segmentations. These observations further motivate the use of refinement experts to correct fragmented and geometrically inconsistent predictions.

B Gating Ablation Study

To assess the contribution of the proposed area-based routing strategy to the observed performance, we perform an ablation study on the CNN-ROI refinement framework. Two routing strategies are evaluated while keeping the refinement architecture, training protocol, data splits, and evaluation metrics identical.

The **first strategy** is the proposed size-based gating mechanism, which routes fragments according to the empirically derived threshold of 5,402 foreground pixels obtained from the exploratory data analysis (Section 3.1). Fragments with predicted areas below this threshold are assigned to the small-fragment expert, while larger fragments are assigned to the large-fragment expert.

Since expert assignment at inference time is determined from the MedSAM-predicted fragment area rather than the ground-truth area, prediction errors may cause a fragment to be routed to a different expert than it would be under ground-truth-based routing. We therefore quantify this discrepancy by comparing the expert assignments obtained from MedSAM-predicted and ground-truth fragment sizes. As shown in Table 7, only 2.899% of fragments are assigned to a different expert overall, with similarly low rates across the training, validation, and test splits. This indicates that the MedSAM-derived fragment area provides a reliable proxy for the ground-truth size when applying the proposed gating rule.

Table 7. Misrouting rate when expert selection is based on the MedSAM-predicted fragment size rather than the ground-truth fragment size. The routing threshold is 5,402 pixels at 1024×1024 resolution, corresponding to a relative area of 0.005152.

Split	Misrouted fragments	Total fragments	Misrouting rate (%)
Train	6121	211926	2.888
Val	810	26544	3.052
Test	748	26457	2.827
Overall	7679	264927	**2.899**

The **second strategy** is a random-routing baseline, which assigns fragments to the two experts uniformly at random while preserving the same small-to-large expert ratio as the proposed method. Consequently, both experts are trained on datasets of identical size, but without any relationship between fragment characteristics and expert assignment. To assess the robustness of the random baseline, experiments are repeated with three different random seeds (7, 42, and 99).

Comparing these two routing strategies isolates the effect of the gating mechanism itself. Since the refinement architecture, optimization settings, and training data are otherwise kept unchanged, differences in performance primarily reflect the effect of routing fragments according to the proposed data-driven criterion versus assigning them randomly.

The ablation is conducted only on the CNN-ROI variant because the routing mechanism is independent of the refinement architecture family. Evaluating the gating strategy on a single refinement architecture isolates the effect of routing while avoiding the substantial computational cost of retraining all refinement models.

Table 8. Gating ablation of the CNN-ROI refinement framework. The proposed area-based gating threshold (5,402 foreground pixels) is compared with random expert assignment while preserving the same expert allocation ratio. Random routing is evaluated using three random seeds (7, 42, and 99) to assess robustness. Best values are shown in **bold**.

Routing strategy	DSC ↑	IoU ↑	HD95 ↓	ASSD ↓
Area-based (5,402 px)	**0.7935**	**0.6702**	36.53	10.28
Random (seed 7)	0.7910	0.6674	35.69	10.06
Random (seed 42)	0.7912	0.6675	**35.44**	**10.04**
Random (seed 99)	0.7919	0.6687	36.36	10.18
Random (mean ± std)	0.7914 ± 0.0005	0.6679 ± 0.0007	35.83 ± 0.48	10.09 ± 0.08

Table 8 summarizes the results. Overall, the proposed area-based routing achieves higher overlap-based performance than the three random-routing baselines, while providing a deterministic and interpretable routing strategy. The proposed gating obtains a DSC of 0.7935 and an IoU of 0.6702, compared with 0.7914 ± 0.0005 and 0.6679 ± 0.0007, respectively, across the three random-routing seeds. This corresponds to improvements of 0.0021 DSC and 0.0023 IoU over the random-routing mean. Moreover, the random-routing results remain consistently below the proposed gating for both overlap metrics across all three seeds, suggesting that the observed advantage is not attributable to a single unfavorable random assignment. The boundary-based metrics show a different trend. Random routing achieves lower HD95 and ASSD on average, with 35.83 ± 0.48 and 10.09 ± 0.08, respectively, compared with 36.53 and 10.28 for the area-based strategy. Hence, the proposed gating does not provide a consistent advantage across all evaluation metrics. Nevertheless, its consistent advantage in DSC and IoU across the evaluated random seeds indicates that routing fragments according to their size can improve region-overlap performance relative to random expert assignment.

The aggregate metrics in Table 8 are computed over the full test set, which is heavily skewed toward large fragments (24,935 of 26,457 fragments, 94.25%); as a result, the reported improvements largely reflect performance on large fragments, where the choice of routing strategy has little effect. To examine whether the benefit of area-based routing is uniform across fragment sizes or concentrated in a specific regime, we plot Dice score and failure rate (DSC < 0.5) as a function of ground-truth fragment size in Figures 7 and 8. Both figures reveal that the advantage of area-based routing is concentrated almost entirely within the small-

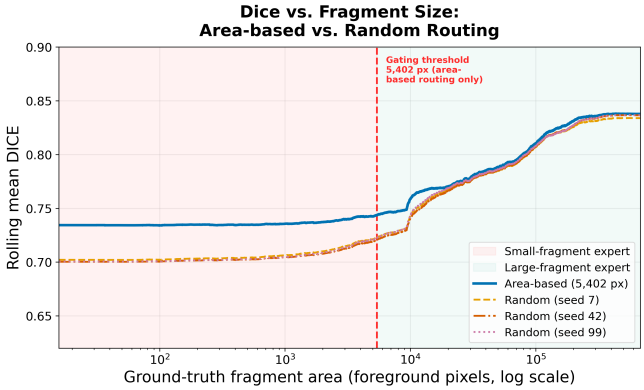


Fig. 7. Mean Dice score as a function of ground-truth fragment size, for the proposed area-based routing strategy versus three random-routing baselines (seeds 7, 42, 99). The dashed line marks the 5,402-pixel gating threshold; shaded regions indicate which expert a fragment of that size would be routed to under area-based routing (random routing ignores size entirely). Area-based routing achieves a consistently higher Dice score than all three random baselines in the small-fragment region (0.710 vs. 0.665–0.668), while the two strategies converge for large fragments.

fragment region: area-based routing achieves a DSC of 0.710 versus 0.665–0.668 for the random baselines, and a failure rate of 11.8% versus 16.0–16.8%, an effect roughly an order of magnitude larger than the aggregate improvement in Table 8. In the large-fragment region, the four curves are nearly indistinguishable. This indicates that the aggregate metrics substantially understate the benefit of the proposed gating mechanism precisely where it is designed to matter most, namely on the small, harder-to-segment fragments that the size-based expert specialization targets. Combined with the pronounced, size-concentrated advantage shown in Figures 7 and 8, and its deterministic and interpretable nature, these results support the proposed area-based criterion as a principled routing strategy for expert specialization.

C Experimental Setup Details

MedSAM First Stage. The MedSAM ViT-B image encoder is run once per image, producing an image-level embedding of shape $256 \times 64 \times 64$ stored in float16. The mask decoder produces low-resolution 256×256 logits per fragment, which are passed through a sigmoid to obtain probabilities, upsampled to 1024×1024 via bilinear interpolation, and binarized at a threshold of 0.5. Per-image outputs, binary masks (`.npz`), image embeddings (`.npy`), and metadata (`.jsonl`), are cached to disk before the second stage.

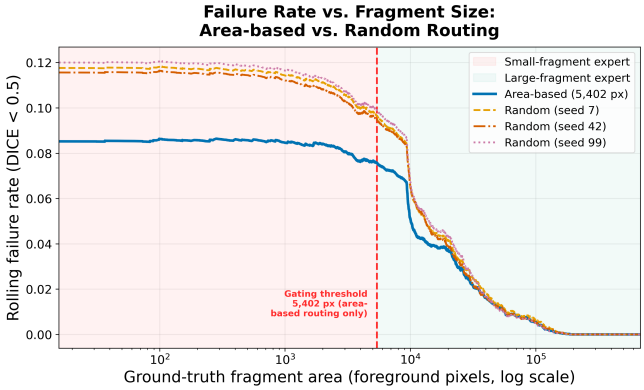


Fig. 8. Fragment-level failure rate (Dice < 0.5) as a function of ground-truth fragment size, for the proposed area-based routing strategy versus three random-routing baselines (seeds 7, 42, 99). Area-based routing yields a consistently lower failure rate than random routing throughout the small-fragment region (11.8% vs. 16.0–16.8%), with the gap closing almost entirely once fragments exceed the 5,402-pixel gating threshold.

D CNN Refinement Architecture and Training Details

Input Representation. The coarse 1024×1024 MedSAM mask is downsampled via nearest-neighbour interpolation to 64×64 and used to derive two channels: the downsampled binary mask itself and a signed distance function (SDF) computed from it via `scipy.ndimage.distance_transform_edt`. The SDF is obtained as the distance transform outside the mask minus the distance transform inside, normalised per instance by its maximum absolute value to lie in $[-1, 1]$. Both channels are concatenated with the $256 \times 64 \times 64$ MedSAM image embedding, yielding an input tensor of shape $258 \times 64 \times 64$.

Architecture. The network consists of two Conv→BN→ReLU blocks reducing channels from 258 to 128 and then to 64 (kernel size 3), followed by a 1×1 convolution and sigmoid activation. The output is produced at 64×64 resolution. For `cnnNoROI`, this output is upsampled to 448×448 to match the ground-truth mask resolution; `cnnROI` instead retains the raw 64×64 output and computes the loss at that resolution (see main text).

Optimisation. Training uses AdamW with a learning rate of 10^{-4} and weight decay of 10^{-4}, a batch size of 64, for up to 40 epochs with early stopping after 10 epochs without validation-loss improvement. To balance the small-fragment and large-fragment experts' training sets, the large-fragment set is subsampled to 17,000 fragments (against ∼17,335 small fragments), stratified to preserve the SA/LI/RI class proportions.

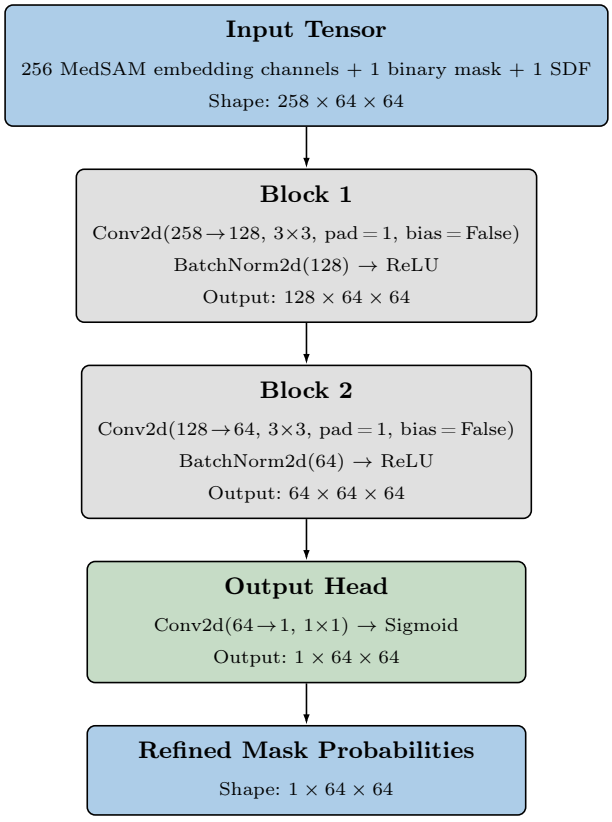


Fig. 9. Architecture of the CNN-based refinement networks. The models take a $258 \times 64 \times 64$ input formed by concatenating the MedSAM image embedding with a binary mask channel and a signed distance function channel. Two convolutional blocks progressively reduce the channel dimension, followed by a 1×1 output head with sigmoid activation producing a $1 \times 64 \times 64$ refined mask.

E Flow Matching Refinement Approach

FlowSDF Architecture. The backbone is a UNet that takes a 1-channel noisy SDF as input and outputs a 1-channel predicted velocity field through a standard encoder-decoder with skip connections, using channel multipliers $(1, 1, 2, 2, 4, 4)$ across resolution levels and self-attention at feature resolutions 16×16 and 8×8. Image conditioning is handled by a separate RRDB network comprising 12 Residual-in-Residual Dense Blocks, which encodes the 257-channel conditioning tensor and injects its features into the first block of the UNet encoder only. During inference, the routed expert begins from a noisy Gaussian sample and repeatedly applies the learned conditional vector field to determine the direction of "flow" toward the target structure, numerically integrated via Ordinary Differential Equations (ODEs) integration over 30 steps with a single stochastic sample, which incrementally transforms the noisy initialization into a coherent segmentation mask. The resulting SDF is binarized at a threshold of 0.03 to produce the refined binary mask, which is then upsampled and saved in a MedSAM-compatible format.

FlowSDF Implementation. The original FlowSDF workflow [1] trains a single image-conditioned model on preprocessed datasets where SDF masks are already saved and later samples full segmentation masks from raw image conditioning. However, in our case, the refinement pipeline adapts FlowSDF to the MedSAM setting by separately training the $\text{expert}_{\text{small}}$ and $\text{expert}_{\text{large}}$ models on gated PENGWIN fragments, using precomputed MedSAM image embeddings plus the coarse MedSAM mask as the conditioning input and computing GT fragment SDF targets on the fly. This adaptation is needed due to the task not being generic image-to-mask generation, since MedSAM already provides strong image features and initial fragment masks, and the goal is to refine those routed fragment predictions in a way that respects fragment-size specialisation, then return results in the existing MedSAM-style mask format for evaluation and integration. Moreover, the FlowSDF refinement approach currently includes expert routing, MedSAM embedding conditioning and SDF refinement, but, as opposed to the CNN variants, does not include any ROI-alignment behaviour.

FlowSDF Loss The model is trained with a conditional Flow Matching loss that learns the vector field between noisy and target SDF distributions:

$$\mathcal{L}_{\text{CFM}}(\theta) = \mathbb{E}_{t\sim\mathcal{U}[0,1]}\mathbb{E}_{(m_1,x)\sim q(m|x)} \\ \mathbb{E}_{m_t\sim p_t(\cdot|m_1,x)}\left[\frac{1}{2}\left\|v_{\theta,t}(m_t,x) - u_t(m \mid m_1,x)\right\|^2\right]$$

FlowSDF Inference Experiments The inference process of the FlowSDF model is dependent on the number of ODE steps, a higher number resulting in smoother edges. In alignment with the paper proposing the model, we started with 4 steps, which yielded a significant improvement compared to the MedSAM baseline. Moreover, we experimented with 10 and 30 ODE steps for the FlowSDF models

trained on 20 epochs, both of which showed promising qualitative improvements in selected examples, and with 40 ODE steps for the FlowSDF models trained on 40 epochs. However, inference becomes substantially more expensive as the number of ODE steps increases: for example, 30 ODE steps required approximately 7 hours on an H100 GPU. Due to this computational cost, the extent of experimentation was limited. Figures 10–12 show the best results of each inference setup, highlighting how increasing the number of steps leads to the refinement approach identifying increasingly more complex anatomical shapes that were initially completely misrepresented by MedSAM.

Table 9. FlowSDF inference time for each ODE-step count configuration. Each configuration was evaluated on the test set of 5,000 images using a single NVIDIA H100 GPU.

ODE steps	Total time	Avg. time per fragment (s)
4	0:53:09	0.638
10	2:22:01	1.704
30	7:26:34	5.359
40	9:57:27	7.169

F Training-Time Comparison

Table 10 summarizes the observed wall-clock training times. The small- and large-fragment experts were trained independently, with a maximum budget of 40 epochs and one GPU per training job; epoch counts are therefore reported as small/large. The `cnnNoROI` and FlowSDF small-fragment experts early-stopped after 38 and 25 epochs, respectively, whereas the remaining experts completed all 40 epochs. The reported MedSAM fine-tuning time is an estimate because the original run was not fully logged and used a different, non-comparable GPU. We therefore extrapolate the 20-epoch runtime from the mean duration of the first four epochs of a separate fine-tuning run (approximately 3:38:06 per epoch). Consequently, the timings provide an indicative comparison rather than a controlled runtime benchmark.

Table 10. Training compute for the expert models.

Model	GPU	Train time (h)	Epochs (small/large)
cnnROI	A100 40GB	8:57:00	40 / 40
cnnNoROI	A100 40GB	7:35:28	38 / 40
FlowSDF	H100 80GB	11:11:21	25 / 40
MedSAM (fine-tuned)	H100 80GB	∼72:06:00	20

Fig. 10. Best result of the FlowSDF refinement approach with 4 ODE steps (anatomical class: LI; DSC improvement: +0.575). The model was trained for 20 epochs.

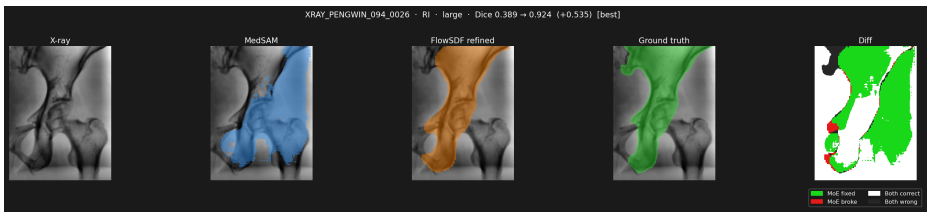


Fig. 11. Best result of the FlowSDF refinement approach with 10 ODE steps (anatomical class: RI; DSC improvement: +0.535). The model was trained for 20 epochs.

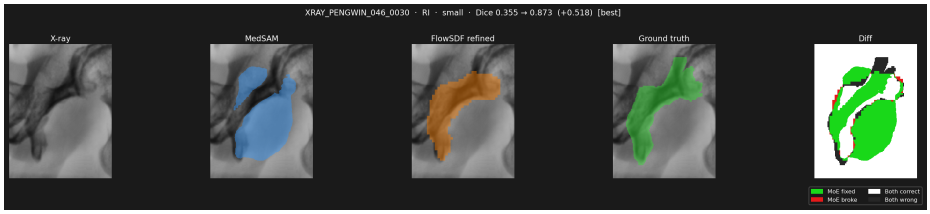


Fig. 12. Best result of the FlowSDF refinement approach with 30 ODE steps (anatomical class: RI; DSC improvement: +0.518). The model was trained for 20 epochs.

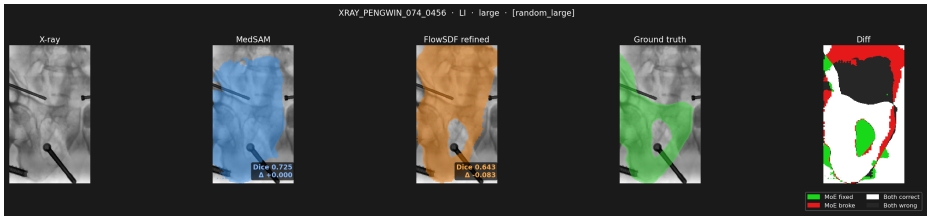


Fig. 13. Best result of the FlowSDF refinement approach with 40 ODE steps (anatomical class: LI; DSC improvement: +0.083). The model was trained for 40 epochs.

G Results

In Table 11 we report the quantitative performance of each refinement strategy on the held-out test split, comparing overlap and boundary metrics across anatomical classes and fragment-size groups. To complement the main paper, Table 12 provides a detailed comparison between the `cnnNoROI` and `cnnROI` variants, highlighting the effect of incorporating ROI-based refinement. Finally, Table 13 presents a detailed boundary-metric comparison between `cnnROI` and FlowSDF, illustrating the substantial improvements achieved by the generative refinement model in terms of HD95 and ASSD.

Table 11. Segmentation performance of all models across all evaluation groups. Results are reported for five models - MedSAM (baseline), cnnNoROI, cnnROI, FlowSDF, and fine-tuned MedSAM (FT-MedSAM) - over six group stratifications: overall, three anatomical classes (SA, LI, RI), and two fragment-size bins (Small, Large). Metrics are Dice Similarity Coefficient (DSC, ↑), Intersection over Union (IoU, ↑), 95th percentile Hausdorff Distance (HD95, ↓), and Average Symmetric Surface Distance (ASSD, ↓), both surface distances in pixels. All models are evaluated on the held-out test split (5 000 images, 26 457 fragments). **Bold** denotes the best-performing model within each group–metric combination.

	DSC↑					IoU↑				
Group	MedSAM	cnnNoROI	cnnROI	FlowSDF	FT-MedSAM	MedSAM	cnnNoROI	cnnROI	FlowSDF	FT-MedSAM
Overall										
All Classes	0.7187	0.7985	0.7935	0.9083	**0.9107**	0.5751	0.6789	0.6702	0.8454	**0.8462**
By anatomical class										
SA	0.7194	0.7743	0.7696	**0.8808**	0.8755	0.5753	0.6441	0.6368	**0.8031**	0.7904
LI	0.7153	0.8059	0.8006	0.9175	**0.9231**	0.5708	0.6889	0.6794	0.8587	**0.8656**
RI	0.7196	0.8087	0.8039	0.9193	**0.9241**	0.5764	0.6943	0.6855	0.8630	**0.8677**
By fragment size										
Small	0.6965	0.7821	0.7781	**0.8809**	0.8803	0.5495	0.6595	0.6510	**0.8025**	0.7985
Large	0.7394	0.8148	0.8088	0.9358	**0.9413**	0.5986	0.6982	0.6894	0.8882	**0.8941**
	HD95 (px)↓					ASSD (px)↓				
Group	MedSAM	cnnNoROI	cnnROI	FlowSDF	FT-MedSAM	MedSAM	cnnNoROI	cnnROI	FlowSDF	FT-MedSAM
Overall										
All Classes	39.85	34.87	36.53	13.99	**13.68**	13.07	9.74	10.28	**3.52**	3.56
By anatomical class										
SA	43.45	38.50	39.88	**19.37**	19.58	14.67	11.89	12.36	**5.12**	5.65
LI	39.27	33.54	35.58	12.04	**11.46**	12.69	8.88	9.51	2.93	**2.78**
RI	37.45	33.55	35.01	12.01	**11.58**	12.28	9.02	9.53	2.93	**2.80**
By fragment size										
Small	22.39	20.11	20.39	**10.11**	10.72	8.03	6.21	6.46	**2.89**	3.03
Large	57.07	49.56	52.67	17.86	**16.64**	18.12	13.24	14.11	4.14	**4.09**

Table 12. `cnnROI` performance broken down by anatomical class and fragment size, with the gain over `cnnNoROI` (Δ) shown. Negative Δ indicates regression. Cell shading reflects the sign and magnitude of each gain.

Group	DSC ↑	IoU ↑	HD95 (px) ↓	ASSD (px) ↓	ΔDSC	ΔIoU
Overall						
Overall	0.7935	0.6702	36.53	10.28	−0.0050	−0.0087
By anatomical class						
SA	0.7696	0.6368	39.88	12.36	−0.0047	−0.0073
LI	0.8006	0.6794	35.58	9.51	−0.0053	−0.0095
RI	0.8039	0.6855	35.01	9.53	−0.0048	−0.0088
By fragment size						
Small	0.7781	0.6510	20.39	6.46	−0.0040	−0.0085
Large	0.8088	0.6894	52.67	14.11	−0.0060	−0.0088

Table 13. `FlowSDF` vs. `cnnROI`—boundary quality (HD95 and ASSD, in pixels). Δ = `FlowSDF` − `cnnROI`; since lower is better, negative Δ favours `FlowSDF` and is highlighted in green. **Bold** indicates the better result per row.

Group	HD95 (px) ↓			ASSD (px) ↓		
	`cnnROI`	`FlowSDF`	Δ	`cnnROI`	`FlowSDF`	Δ
Overall	36.53	**13.99**	−22.54	10.28	**3.52**	−6.76
SA	39.88	**19.37**	−20.51	12.36	**5.12**	−7.24
LI	35.58	**12.04**	−23.54	9.51	**2.93**	−6.58
RI	35.01	**12.01**	−23.00	9.53	**2.93**	−6.60
Small	20.39	**10.11**	−10.28	6.46	**2.89**	−3.57
Large	52.67	**17.86**	−34.81	14.11	**4.14**	−9.97

H Model-Agnostic Experiment

To demonstrate that the proposed refinement framework is model-agnostic, we replace MedSAM with the original SAM ViT-B [4] while keeping the refinement architecture, gating mechanism, and evaluation protocol unchanged. The small-fragment and large-fragment `cnnNoROI` experts are retrained from scratch using SAM generated embeddings and masks.

The SAM-based pipeline achieves an overall DSC of 0.6894 and IoU of 0.5517 as shown on Table 14. Although performance is lower than with MedSAM, due to SAM's lack of medical-domain fine-tuning and the shorter training schedule, the framework operates without architectural modifications. This confirms that the proposed refinement approach can be adapted to different compatible base segmentation models.

I Generalization to a New Anatomy

I.1 Experimental Details

The RAM-H1200 (Rheumatoid Arthritis Modeling–Hand 1200) dataset [18] is a multi-center collection of 1,200 posteroanterior hand radiographs from 291 patients, acquired across six medical institutions in Japan. The dataset was developed to support automated analysis of rheumatoid arthritis (RA) and provides multiple levels of expert annotation, including pixel-level segmentation of 30 anatomical bone structures and bone erosions, as well as joint-level Sharp/van der Heijde (SvdH) scores for bone erosion (BE) and joint-space narrowing (JSN). Specifically, SvdH annotations cover 16 locations per hand for BE and 15 for JSN. All radiographs were standardized to a spatial resolution of 0.175 mm/pixel, with left-hand images horizontally flipped to provide a consistent orientation. Importantly, annotations were independently assessed by at least two readers, resolved through consensus, and subsequently verified by an experienced radiologist. The dataset therefore combines detailed anatomical segmentation with clinically established measures of RA-related structural damage, enabling both image segmentation and disease-severity assessment tasks.

I.2 Zero Shot Evaluation

To evaluate the two frozen experts on RAM-H1200, we first needed a rule for routing each bone fragment to the small- or large-object expert. For this, the frozen FlowSDF checkpoints are used, run with 4 ODE steps. We evaluated zero-shot performance under PENGWIN's own static threshold (5,402 px) first. However, this threshold reflects PENGWIN's pelvic-fracture fragment-area distribution, which sits on a different absolute scale than RAM-H1200's hand-bone fragments - even after preprocessing, the objects themselves are simply smaller. We therefore additionally derived a RAM-H1200-specific threshold using the same methodology PENGWIN's own threshold came from: a rolling MedSAM-dice-vs-fragment-area

Table 14. Segmentation performance of plain SAM (no medical fine-tuning) before and after refinement by the cnnNoROI refinement framework, using identical bounding-box prompts for both. cnnNoROI was trained for 10 epochs (patience 10, large-subsample 17,000). Results are reported over six group stratifications: overall, three anatomical classes (SA, LI, RI), and two fragment-size bins (Small, Large). Metrics are Dice Similarity Coefficient (DSC, ↑), Intersection over Union (IoU, ↑), 95th percentile Hausdorff Distance (HD95, ↓), and Average Symmetric Surface Distance (ASSD, ↓), both surface distances in pixels. Both evaluated on the held-out test split (5 000 images, 26 457 fragments). **Bold** denotes the better value within each group–metric pair.

	DSC↑		IoU↑	
Group	SAM only	SAM + cnnNoROI	SAM only	SAM + cnnNoROI
Overall				
All Classes	0.5709	**0.6894**	0.4270	**0.5517**
By anatomical class				
SA	0.5699	**0.6771**	0.4267	**0.5355**
LI	0.6005	**0.7068**	0.4550	**0.5726**
RI	0.5412	**0.6805**	0.3987	**0.5423**
By fragment size				
Small	0.5904	**0.6869**	0.4484	**0.5506**
Large	0.5513	**0.6918**	0.4056	**0.5527**

	HD95 (px)↓		ASSD (px)↓	
Group	SAM only	SAM + cnnNoROI	SAM only	SAM + cnnNoROI
Overall				
All Classes	**47.86**	51.16	16.52	**16.05**
By anatomical class				
SA	**49.58**	53.81	**17.84**	18.03
LI	**46.62**	48.89	15.60	**14.89**
RI	**47.85**	51.53	16.47	**15.78**
By fragment size				
Small	**28.44**	36.19	**9.74**	12.09
Large	67.29	**66.07**	23.29	**20.00**

analysis that identifies the area at which segmentation quality actually degrades, rather than an arbitrary midpoint of the size distribution. This yielded a threshold of 3,433 px—about 36% lower than PENGWIN's. Since deriving this threshold requires GT masks, which may not always be available (though even a small labeled subset is enough to get a reasonable estimate), we also computed a simpler, label-free alternative: the median area of MedSAM's own predicted masks, which is 3,988 px (lower than PENGWIN's threshold and closer to the derived one). As shown in Tables 15 - 16, routing by this median threshold yields highly comparable results to the derived threshold, suggesting it is a reasonable proxy when GT-based derivation is not feasible. We therefore report results under all three routing conventions: PENGWIN's static 5,402 px threshold, RAM-H1200's dice-derived 3,433 px threshold, and the label-free 3,988 px median threshold.

Routing	Group	DSC↑			IoU↑		
		MedSAM	FlowSDF	Δ	MedSAM	FlowSDF	Δ
PENGWIN thresh.	Small	0.6965	**0.7200**	+0.0235	0.5510	**0.5757**	+0.0247
	Large	**0.8110**	0.7151	−0.0959	**0.6966**	0.5675	−0.1291
Dice-drop thresh.	Small	0.6719	**0.7022**	+0.0303	0.5229	**0.5559**	+0.0330
	Large	**0.8008**	0.7174	−0.0834	**0.6810**	0.5692	−0.1118
Median thresh.	Small	0.6815	**0.7104**	+0.0289	0.5335	**0.5650**	+0.0315
	Large	**0.8057**	0.7176	−0.0881	**0.6882**	0.5698	−0.1184

Table 15. Zero-shot transfer to RAM-H1200: frozen `FlowSDF` experts applied to MedSAM outputs, no retraining. Region overlap (DSC, IoU; ↑). Δ = `FlowSDF` − `MedSAM`; green favours `FlowSDF`, red marks regression. Each routing scheme is compared against its own MedSAM baseline. **Bold** marks the better of MedSAM/FlowSDF per row.

Routing	Group	HD95 (px)↓			ASSD (px)↓		
		MedSAM	FlowSDF	Δ	MedSAM	FlowSDF	Δ
PENGWIN thresh.	Small	23.15	**19.88**	−3.27	8.38	**8.33**	−0.05
	Large	**26.96**	37.66	+10.7	**9.06**	15.08	+6.02
Dice-drop thresh.	Small	22.85	**19.40**	−3.45	8.37	**8.09**	−0.28
	Large	**26.21**	35.06	+8.85	**8.89**	14.32	+5.43
Median thresh.	Small	23.00	**19.50**	−3.50	8.40	**8.15**	−0.25
	Large	**26.43**	35.94	+9.51	**8.92**	14.58	+5.66

Table 16. Zero-shot transfer to RAM-H1200: boundary quality (HD95, ASSD, px; ↓). Δ = `FlowSDF` − `MedSAM`; negative (green) favours `FlowSDF`. The frozen small-fragment expert improves boundaries across anatomy, while the large-fragment expert regresses on the short, elongated hand bones—consistent with regime-specific specialisation. **Bold** marks the better per row.

Qualitatively, Figure 14 shows how the three thresholds yield different routing decisions and, in turn, different refinements. Because the thresholds are ordered PENGWIN > Median > Dice-drop, progressively fewer fragments fall below the cut and are sent to the small expert. This is visible in the proximal phalanges of the middle and ring fingers (teal and green masks). Under the PENGWIN threshold, both bones are routed to the small expert and receive well-refined masks. Under the Median threshold, only one falls to the small expert, producing one refined mask alongside a coarser, more blob-like one. Under the Dice-drop threshold, both bones are handled by the large expert, yielding coarser masks poorly matched to these thin, mid-sized structures.

I.3 Full Training Qualitative Examples

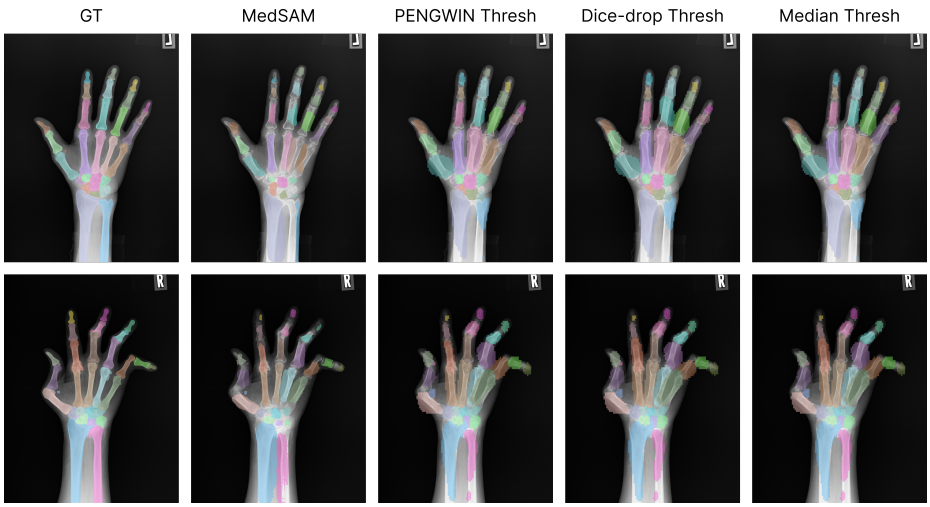


Fig. 14. Qualitative comparison of frozen checkpoints (flowSDF experts trained on PENGWIN). Inference ran on new anatomy samples from RAM-H1200.

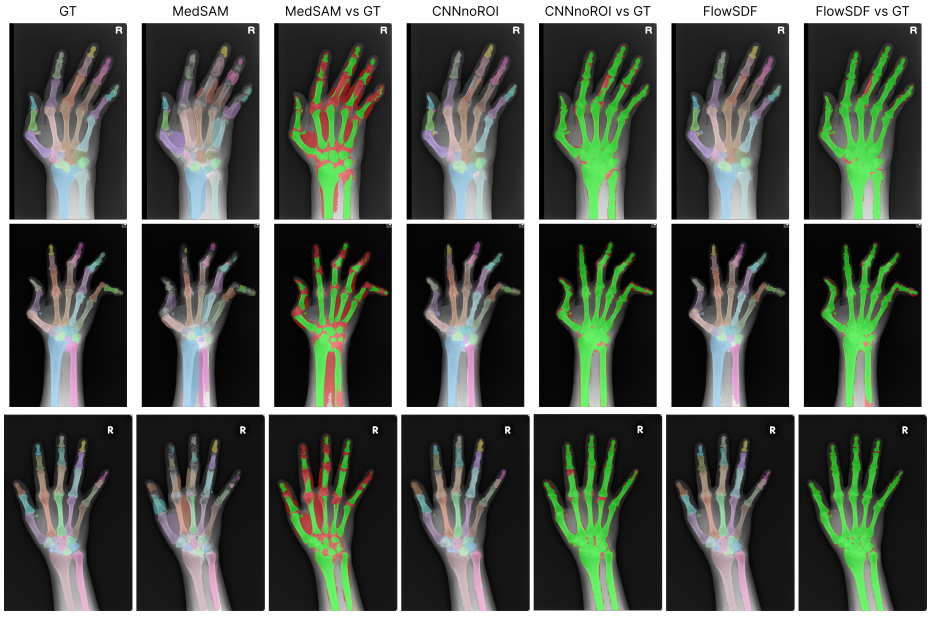


Fig. 15. Qualitative comparison of Ground Truth, MedSAM, cnnNoROI and FlowSDF masks with differences raported to GT.

J Learned Gating Ablation

As an extension to the fixed area-threshold routing used elsewhere in this work, we implement and evaluate a *learned* gating mechanism, in which a small gate network learns to route each fragment to one of two experts jointly with the experts themselves, rather than routing being imposed a priori. Both experts use the same architecture as `cnnNoROI` and are randomly initialized and trained from scratch, so that any resulting specialization emerges from training rather than from warm-starting.

Gate. The gate operates on the same input tensor $x \in \mathbb{R}^{258\times64\times64}$ the experts consume: global average pooling over the spatial dimensions, followed by a two-layer MLP,

$$\bar{x} = \mathrm{GAP}(x) \in \mathbb{R}^{258}, \tag{1}$$

$$\ell(x) = W_2\,\mathrm{ReLU}(W_1\bar{x} + b_1) + b_2 \in \mathbb{R}^2, \tag{2}$$

with $W_1 \in \mathbb{R}^{64\times258}$, $W_2 \in \mathbb{R}^{2\times64}$. The gate is given *no* explicit size or shape descriptors — it can only route on whatever signal is recoverable from the pooled embedding, mask, and SDF channels. During training only, we add Gaussian noise to the logits before the softmax, following in simplified form the noisy gating of Shazeer et al. [15]: we use a fixed noise scale rather than their learned, input-dependent scale, and, with only two experts and a dense (non-top-k) mixture, the noise serves purely to encourage exploration:

$$\tilde{\ell}(x) = \ell(x) + \varepsilon \cdot z, \qquad z \sim \mathcal{N}(0, I_2), \qquad \varepsilon = 0.3, \tag{3}$$

disabled at evaluation time. Gate weights are $g(x) = \mathrm{softmax}(\ell(x))$, with $g_0(x) + g_1(x) = 1$ (using $\tilde{\ell}$ during training). At training time, both experts are evaluated on every fragment and blended,

$$\hat{y}(x) = g_0(x)\,y_0(x) + g_1(x)\,y_1(x), \tag{4}$$

and this blended prediction is what the training loss is computed against.

Load-balancing loss. Because both experts are freshly initialized, an unconstrained gate can collapse onto a single expert early in training. We add the importance-balancing auxiliary loss of Shazeer et al. [15] to the segmentation loss:

$$\mathcal{L} = \mathcal{L}_{\mathrm{seg}}(\hat{y}, y_{\mathrm{gt}}) + \lambda_{\mathrm{bal}}\,\mathcal{L}_{\mathrm{bal}}(g), \qquad \lambda_{\mathrm{bal}} = 0.1 \tag{5}$$

where $\mathcal{L}_{\mathrm{seg}}$ is the same segmentation loss used throughout this work. For a batch of B fragments, define the *importance* of expert k as the total gate mass it receives across the batch,

$$I_k = \sum_{i=1}^{B} g_k(x_i), \qquad k \in \{0, 1\}, \tag{6}$$

and penalize the squared coefficient of variation of $I = (I_0, I_1)$ across experts,

$$\mathcal{L}_{\text{bal}} = \left(\frac{\text{std}(I)}{\text{mean}(I)} \right)^2 . \tag{7}$$

For $n = 2$ this reduces to

$$\mathcal{L}_{\text{bal}} \propto \left(\frac{I_0 - I_1}{I_0 + I_1} \right)^2 , \tag{8}$$

a normalized squared imbalance that vanishes if and only if both experts receive equal aggregate gate mass. This term is what keeps both experts active under fresh initialization; we observed no collapse: the mean gate weight of the minority expert remained above 15% throughout training.

Training and data. Training otherwise follows the same setup as `cnnNoROI` (optimizer, learning rate, batch size, epoch budget, early stopping). The one substantive difference is the training data itself: both experts and the gate are trained on the *union* of the fragments the two `cnnNoROI` experts each see separately — no area-threshold routing is applied to the dataloader, since specialization must emerge rather than be imposed. We note the validation and test fragment populations are class-imbalanced by size (large fragments substantially outnumber small ones), which should be kept in mind when interpreting size-stratified results.

Inference. At test time, noisy gating is disabled and we evaluate two routing modes from the same trained checkpoint:

- **Hard routing:** $k^* = \arg\max_k g_k(x)$; the output is $y_{k^*}(x)$ from the selected expert alone. This matches the one-expert-per-fragment inference structure of the fixed-threshold baseline.
- **Soft routing:** $\hat{y}(x) = g_0(x)y_0(x) + g_1(x)y_1(x)$, identical in form to the training-time mixture.

Results Table 17 compares the learned hard- and soft-routing modes with the frozen MedSAM baseline and the static area-gated `cnnNoROI` model. We report region overlap using DSC and IoU (↑) and boundary accuracy using HD95 and ASSD in pixels (↓), both overall and stratified by anatomical class and fragment size. This comparison isolates whether the gate can recover a useful routing signal from the pooled MedSAM embedding, mask, and SDF channels without being given explicit size or shape descriptors.

The learned gate yields its strongest region-overlap results when the soft mixture is retained at inference. Overall, soft routing reaches a DSC of 0.8017 and an IoU of 0.6832, modest improvements of 0.0032 and 0.0043 over the static `cnnNoROI` baseline. It also obtains the highest DSC and IoU for every anatomical class and both fragment-size groups. Hard routing remains close to the static baseline overall (0.7987 DSC and 0.6790 IoU), but trails soft routing in every overlap comparison and notably underperforms the static gate for small fragments.

Table 17. Segmentation performance in the learned-gating ablation. **Bold** denotes the best-performing model within each group–metric combination.

	DSC↑				IoU↑			
Group	MedSAM	cnnNoROI	hardGate	softGate	MedSAM	cnnNoROI	hardGate	softGate
Overall								
All Classes	0.7187	0.7985	0.7987	**0.8017**	0.5751	0.6789	0.6790	**0.6832**
By anatomical class								
SA	0.7194	0.7743	0.7736	**0.7780**	0.5753	0.6441	0.6433	**0.6489**
LI	0.7153	0.8059	0.8067	**0.8094**	0.5708	0.6889	0.6898	**0.6936**
RI	0.7196	0.8087	0.8091	**0.8115**	0.5764	0.6943	0.6944	**0.6978**
By fragment size								
Small	0.6965	0.7821	0.7793	**0.7849**	0.5495	0.6595	0.6510	**0.6628**
Large	0.7394	0.8148	0.8181	**0.8186**	0.5986	0.6982	0.7028	**0.7035**
	HD95 (px)↓				ASSD (px)↓			
Group	MedSAM	cnnNoROI	hardGate	softGate	MedSAM	cnnNoROI	hardGate	softGate
Overall								
All Classes	39.85	**34.87**	36.38	35.91	13.07	**9.74**	10.09	9.99
By anatomical class								
SA	43.45	**38.50**	39.71	39.15	14.67	**11.89**	12.11	12.02
LI	39.27	**33.54**	35.24	34.95	12.69	**8.88**	9.41	9.29
RI	37.45	**33.55**	35.08	34.48	12.28	**9.02**	9.30	9.20
By fragment size								
Small	22.39	**20.11**	22.66	21.65	8.03	**6.21**	6.87	6.64
Large	57.07	**49.56**	50.06	50.11	18.12	**13.24**	13.31	13.32

This pattern is consistent with the training formulation: the segmentation loss is optimized on a weighted combination of both expert predictions, whereas hard inference retains only the highest-weighted expert.

The boundary metrics provide a more cautious result. Overall, soft routing improves on hard routing (HD95 35.91 vs. 36.38 px; ASSD 9.99 vs. 10.09 px), and the same ordering holds for every anatomical class and for small fragments. For large fragments, however, hard routing is marginally better than soft routing (HD95 50.06 vs. 50.11 px; ASSD 13.31 vs. 13.32 px). Neither learned variant surpasses the static gate on boundary accuracy in any reported group; overall, `cnnNoROI` obtains 34.87 px HD95 and 9.74 px ASSD. The absence of gate collapse and the consistent overlap gains suggest that the pooled representation contains a useful routing signal, but the learned mixture benefits region overlap more than precise boundary localization. These results motivate further study of learned gates incorporating explicit geometry, boundary complexity, or uncertainty cues.

Representative soft-routing segmentation outputs are shown in Figure 16.

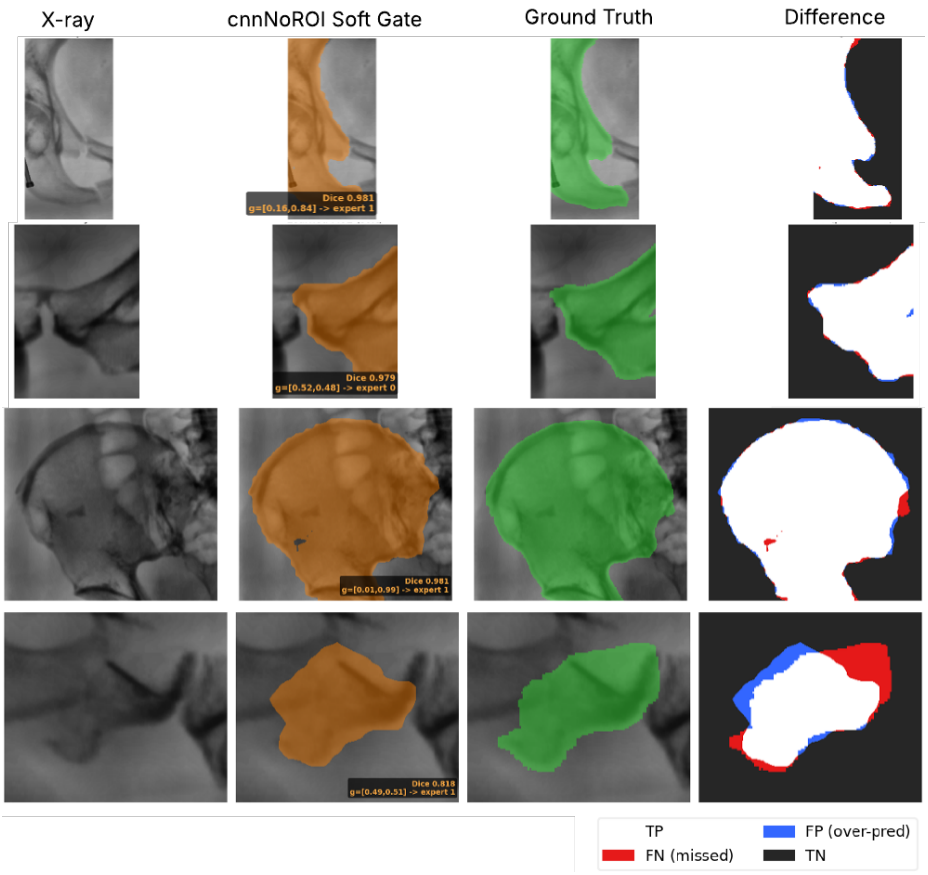


Fig. 16. Example segmentations produced by soft learned gating, which blends the two expert predictions using their learned gate weights.