Fine-Tuned SAM and MedSAM for Automatic Bone Fracture Segmentation in Radiographs

Muhammad Ilyas[1]*, Muhammad Javed Iqbal[1], Moavia Hassan[1], Syed Mushhad M. Gilani[2]
[1]Department of Computer Science, University of Engineering and Technology, Taxila, Pakistan.
[2]Department of Computer Science, University of Agriculture, Faisalabad, Pakistan.
***Correspondence**: ranailyas50137@gmail.com

Citation | Ilyas. M, Iqbal. M. J, Hassan. M, Gilani. S. M. M, "Fine-Tuned SAM and MedSAM for Automatic Bone Fracture Segmentation in Radiographs", IJIST, Vol. 8 Issue. 11 pp 4212-4234, November 2026
Received | Oct 25, 2026 **Revised |** Nov 10, 2026 **Accepted |** Nov 12, 2026 **Published |** Nov 17, 2026.

Bone fractures are common in emergency radiology, and outlining them at pixel level supports measurement and treatment planning. Existing studies mostly focus on classification or box detection, and segmentation networks need many pixel labels for targets that cover less than 1% of a radiograph. This study proposes a two-stage pipeline in which an RT-DETR detector proposes fracture boxes and a box-prompted foundation model, the Segment Anything Model (SAM) or MedSAM, delineates each fracture. Only the mask decoder was fine-tuned on the 717 fractured radiographs of the public FracAtlas dataset (573 training, 86 validation and 58 test images), and the detector threshold was chosen on validation Dice. The detector reached a test precision of 0.709, recall of 0.557 and F1-score of 0.624. With ground-truth boxes, fine-tuning raised mean Dice from 0.701 to 0.843 for SAM (IoU 0.735) and from 0.736 to 0.833 for MedSAM (IoU 0.720), both with $p < 0.001$, and the two fine-tuned models did not differ significantly ($p = 0.26$). In the automatic pipeline, the transformer detector and the Dice-based threshold raised mean Dice from 0.465 to 0.558 (IoU 0.450) and reduced the test images without a prediction from 17 to 3. The pipeline matched an end-to-end YOLO11-seg model in mean Dice (0.558 against 0.540, $p = 0.97$) with a lower median HD95 (12.9 against 17.2 pixels). In practice, one box drawn by a clinician or proposed by a detector yields an accurate fracture outline for measurement, and detection recall is the main remaining limit.

Keywords: Bone Fracture Segmentation; Segment Anything Model; MedSAM; RT-DETR; FracAtlas

Introduction:

Bone fractures are among the most common presentations in emergency radiology, and plain radiography is the first imaging test. Reading radiographs takes time, and subtle or non-displaced fractures are often missed when readers are tired or less experienced. Deep learning models trained on radiographs detect fractures accurately and help clinicians find fractures that would otherwise be missed [1][2], and a systematic review and meta-analysis found that the diagnostic performance of artificial intelligence in fracture detection is comparable to that of clinicians [3].

Most of this work reports image-level classification or bounding-box detection. Pixel-level segmentation also gives the area and shape of the fracture, which is useful for measurement and follow-up. For example, segmentation of the radius and ulna on radiographs of distal radius fractures has been used to measure radial inclination, tilt and height automatically [4], and rib fractures have been segmented on CT with an attention-based U-Net [5]. Segmentation has nevertheless been studied less than detection, because pixel labels are expensive to produce. The FracAtlas dataset [6] provides polygon masks for fractures in several body regions, so segmentation can be measured on public data. Two problems make FracAtlas difficult: only 717 of its radiographs have masks, and a fracture usually covers less than 1% of the image.

Promptable foundation models offer another route. The Segment Anything Model (SAM) [7] was trained on more than one billion masks of natural images and returns a mask for any region indicated by a point or a box. MedSAM [8] adapted SAM to medical images with a large set of image-mask pairs from many modalities. Because the image encoder is already pretrained and the prompt only has to locate the object, both models need far less task-specific training than a network trained from scratch. Reviews and evaluations of SAM on medical images show that its accuracy depends strongly on the task and is often lower than that of specialized networks when it is used without adaptation [9][10][11], and fine-tuning studies show that adapting the model to the target data closes much of this gap [12].

Research Gap:

The existing work on SAM and MedSAM leaves four gaps that are relevant to fracture radiographs. First, most evaluations use CT, MRI, ultrasound, endoscopy or dermoscopy, where the targets are large, thick and high-contrast [9][10][11]. Even for bone, a recent zero-shot study segmented whole bones in CT rather than fracture lines [13]. Radiographic fractures are thin, low in contrast and small, and it is not known how well these models delineate them. Second, most studies prompt the model with ground-truth boxes or points [9][10]. This shows the best case, but in clinical use no expert box is available unless a reader draws one, so the reported accuracy overstates what an automatic system would achieve. Third, fine-tuning studies [12][14] show that adaptation helps, but they did not include fracture radiographs.

Fourth, pipelines that pass detector boxes to SAM exist [15], but they use the zero-shot model and do not study how the detector threshold should be chosen for segmentation. This choice matters, because a missed fracture gives a Dice of zero for the image, while a false box only adds a few false-positive pixels. The effect of the detector threshold on the final segmentation has rarely been measured. On FracAtlas, [16] evaluated zero-shot MedSAM with ground-truth boxes and used its masks to refine training labels, but they did not fine-tune it or prompt it with detector boxes, and they named detector-generated prompts as future work.

Research Objectives:

The study has four objectives. Each is linked to the experiment and the metrics used to assess it.

Objective 1: to measure how accurately zero-shot SAM and MedSAM segment fractures on radiographs when given ground-truth boxes. Experiment: zero-shot inference on the test set (see Segmentation with ground-truth boxes). Metrics: Dice, IoU, pixel precision and recall and HD95 with 95% bootstrap confidence intervals (Tables 2 and 3).

Objective 2: to measure the gain from fine-tuning only the mask decoder, and to test whether medical pretraining still helps after fine-tuning. Experiment: decoder fine-tuning of both models with the same settings (see Fine-tuning of the mask decoder and Segmentation with ground-truth boxes). Metrics: paired Wilcoxon signed-rank tests with Holm correction, bootstrap confidence intervals of the differences and effect sizes (Table 4).

Objective 3: to build a fully automatic pipeline and to find which choices in the detection stage limit the segmentation. Experiment: YOLO11 and RT-DETR detectors, detector selection on validation F1 and threshold selection on validation Dice (see Detection stage, Detector threshold for segmentation, Detection and Automatic pipeline). Metrics: instance precision, recall, F1-score and AP50 for detection (Table 1), and mean Dice and the number of empty predictions for each combination of detector and threshold rule (Table 5).

Objective 4: to compare the automatic pipeline with an end-to-end segmentation model and to test its robustness. Experiment: a YOLO11m-seg baseline trained on the same images, shifted ground-truth boxes and fracture-size tertiles (see Automatic pipeline, Robustness to Imperfect Boxes and Fracture Size). Metrics: mean, median and global Dice, HD95 and empty predictions (Tables 2 and 4, Figures 9 and 10).

Novelty Statement and Contributions:

To the best of our knowledge, this is the first study on FracAtlas that fine-tunes the SAM and MedSAM mask decoders for fracture segmentation and prompts them with boxes from a transformer detector. The new element is the combination of three parts: a fine-tuned box-prompted decoder that is trained to tolerate shifted boxes, an RT-DETR detector that supplies the boxes automatically, and a detector threshold chosen on the validation Dice of the final masks instead of on detection F1. Together they raise automatic mean Dice from 0.465 to 0.558 and show how much of the remaining error comes from detection rather than from segmentation. This study extends earlier work on the same dataset [17] with a different segmentation approach. Its contributions are as follows:

A benchmark of SAM and MedSAM for fracture segmentation on FracAtlas, with zero-shot and fine-tuned mask decoders, underground-truth and automatic box prompts, on one fixed split with bootstrap confidence intervals and paired significance tests.

A measurement showing that fine-tuning the mask decoder alone raises mean Dice by 0.14 for SAM and 0.10 for MedSAM and that the advantage of medical pretraining disappears after fine-tuning.

An analysis of the automatic pipeline showing that a transformer detector and a detector threshold chosen on validation Dice together raise mean Dice from 0.465 to 0.558.

A comparison with an end-to-end YOLO11-seg baseline trained on the same images, together with an analysis of robustness to imperfect boxes and of performance by fracture size.

The rest of the paper is organized as follows. The Related Work reviews fracture detection, fracture segmentation, promptable foundation models and detector ensembling. Materials and Methods describes the workflow, the dataset, the detection stage, the segmentation models, decoder fine-tuning, threshold selection, the baseline and the evaluation protocol. The Results report the detection, segmentation, robustness and fracture-size results. The Discussion examines the findings, compares them with published studies, and describes the clinical implications, the limitations and the recommendations for future research. The Conclusion summarizes the main findings.

Related Work:

Fracture Detection in Radiographs:

Before deep networks, image classification relied on hand-crafted local descriptors. For example, KAZE features encoded as a bag of visual words with a two-step classifier were used to classify high-resolution remote sensing images [18]. CNNs trained on large labelled sets detect wrist fractures with an accuracy close to that of experts and reduce reader errors [1]. Most published systems perform classification or detection, and pixel-level outputs are rare [2][3]. Transformer detectors such as DETR [19] removed hand-crafted anchors and non-maximum suppression, and RT-DETR [20] reached real-time speed with an accuracy similar to YOLO models of comparable size on natural images.

Single-stage YOLO detectors [21] are the most common choice for fracture localisation because they are fast and easy to train. Earlier, [22] trained a Faster R-CNN detector on wrist radiographs and localised fractures with high sensitivity on both frontal and lateral views. The release of the GRAZPEDWRI-DX dataset, with 20,327 paediatric wrist radiographs annotated with bounding boxes [23], made YOLO the standard approach for wrist fractures, and YOLOv8 reached high sensitivity on it [24]. Later work changed the backbone or added attention: ResYOLO11 combined ResNet layers with YOLO11 modules and reported mAP50 values between 0.970 and 0.986 on GRAZPEDWRI-DX across model sizes [25].

Results on FracAtlas are much lower, because the dataset covers the hand, leg, hip and shoulder and contains only 717 fractured images. The dataset paper reported an mAP50 of 0.562 for YOLOv8s on its validation set [6]. [26] added dynamic snake convolution and a weighted channel attention module to YOLOv8 (WCAY) and reached an mAP50 of 0.567 on a random 70/20/10 split, against 0.479 for their YOLOv8s baseline. [27] reported a mAP of 0.862 on FracAtlas for an attention-based YOLOv7 (YOLOv7-ATT) in a generalisation experiment. These studies used different splits and evaluation settings, so their numbers are not directly comparable, and transformer detectors such as RT-DETR [20] have rarely been tested on multi-region fracture radiographs.

Fracture Segmentation:

Medical segmentation networks are usually encoder-decoder models based on U-Net [28], trained with overlap losses such as the Dice loss [29] or with the focal loss [30] when the foreground is small. FracAtlas [6] provides instance-level fracture masks for 717 radiographs and reports a YOLOv8 segmentation baseline. Results from different studies on FracAtlas are difficult to compare because they use different splits and preprocessing.

On FracAtlas, [31] compared ResNet50, ResNet101 and MobileNetV3 encoders in DeepLabV3 and reported an IoU above 0.93 for the ResNet-based models on the original images, with strong results also on augmented and cropped subsets. Outside FracAtlas, [32] segmented hip fractures in a collection of 10,308 hip radiographs with YOLOv8 and reached a mean Dice of 0.77, and [33] segmented pelvic fracture fragments in CT with a two-stage 3D U-Net and reported a Dice of 0.986 on the PENGWIN dataset. On CT, [5] segmented rib fractures in axial slices with a composite attention residual U-Net and reported a Dice of 0.533 on the RibFrac test images. [4] segmented the radius and ulna on radiographs of distal radius fractures and derived radial inclination, tilt and height automatically, with intraclass correlations of 0.868 to 0.975 against manual measurement. Self-configuring frameworks such as nnU-Net [34] provide strong baselines for these tasks. [16] benchmarked one-stage, two-stage and transformer instance segmentation models on FracAtlas with five-fold cross-validation, and their final model, a YOLOv8x-seg trained with MedSAM-refined labels, reached a mask mAP50 of 0.63 on a held-out test set.

All of these networks are trained from task-specific pixel labels. Fracture masks are expensive to draw, and a fracture occupies a very small part of a radiograph, so these methods

need either large annotated sets or cropping around the fracture. A segmentation model that reuses a large pretrained encoder and needs only a box to locate the fracture would reduce both problems.

Promptable Foundation Models in Medical Imaging:

SAM [7] combines a Vision Transformer [35] image encoder, pretrained with masked autoencoding [36], with a light prompt encoder and mask decoder. It can segment medical images without training, but its accuracy depends strongly on the modality, the object and the prompt, and box prompts generally work better than point prompts [9][10]. MedSAM [8] retrained the whole model on medical image and mask pairs and improved box-prompted segmentation across many tasks. The organs and lesions in these benchmarks are usually large, thick and high in contrast, unlike radiographic fractures, which are thin, low in contrast and small.

Several ways of adapting SAM to medical images have followed. The Medical SAM Adapter inserts small adapter modules into SAM and trains only these, which improved results over the original model on a range of medical segmentation tasks [14]. Low-rank adaptation [37] is another way to tune a large encoder with few trainable parameters. SAM 2 extended the model to video by adding a memory of earlier frames [38]. Independent evaluations remain mixed: on lumbar spine MRI, MedSAM produced more consistent masks than SAM, but neither model matched a fully trained nnU-Net [39]. A review of SAM in medical imaging found that applying SAM directly does not give satisfactory results on multi-modal and multi-target medical datasets, and summarized the many efforts to adapt it [11]. [12] compared 18 fine-tuning strategies on 17 radiology datasets and found that fine-tuning SAM gives slightly better results than earlier segmentation methods, with parameter-efficient tuning of both the encoder and the decoder working best. In a zero-shot study of bone segmentation in CT, SAM and SAM 2 performed better than medically fine-tuned variants, and a box combined with a centre point was the best prompt [13]. [15] used YOLOv8 to generate boxes for SAM in ultrasound, CT and X-ray images, which removes the need for manual prompts, but the SAM model was not fine-tuned. Most of these evaluations concern CT, MRI, ultrasound or dermoscopy, and few have examined fracture radiographs, where the target is a thin line or a small region inside bone. On FracAtlas, [16] found that zero-shot MedSAM placed its masks near the fracture but underestimated its width and extent, with a mean instance IoU of 0.249 for the boxes supplied with the dataset and 0.317 for tighter boxes derived from the masks. They therefore used MedSAM only to refine training labels within a narrow band around the expert annotation.

Ensembling Detectors:

Weighted Boxes Fusion [40] combines the boxes of different detectors by averaging overlapping boxes, weighted by their confidence. It is commonly used to combine detectors with different error patterns, such as a convolutional detector and a transformer detector.

Standard non-maximum suppression keeps only the highest-scoring box in each group of overlapping boxes. Soft-NMS [41] lowers the scores of overlapping boxes instead of removing them, and Weighted Boxes Fusion averages their coordinates [40]. Fusion is attractive when two detectors have different architectures, such as a convolutional YOLO model and a transformer RT-DETR model, because they may miss different objects. Whether such fusion helps on fracture radiographs with few training images has not been reported, so this study tests it on the validation set before choosing the detector.

Materials and Methods:

The method has two stages. A detector proposes fracture boxes, and each box is passed as a prompt to SAM or MedSAM, which returns a mask. The prediction for an image is the union of these masks. Figure 1 shows the complete workflow in six stages. (1) Dataset preparation: the 717 fractured radiographs are split into training, validation and test sets, and

normal radiographs are set aside for the detector (see Dataset and split). (2) Detector training: YOLO11m and RT-DETR-L are trained on the same images (see Detection stage). (3) Box generation and detector selection: each candidate detector predicts boxes on the validation set, its threshold is set by validation F1, and the candidate with the best validation F1 is kept (see Detection stage). (4) Decoder fine-tuning: the image encoder and prompt encoder of SAM and MedSAM are frozen and only the mask decoder is trained with ground-truth boxes (see Promptable segmentation models and Fine-tuning of the mask decoder). (5) Threshold selection: the full pipeline is run on the validation set and the detector threshold that gives the highest mean Dice is kept (see Detector threshold for segmentation). (6) Evaluation: all models are evaluated once on the test set with the metrics described under Evaluation. Figure 2 follows one test radiograph through the pipeline.



Figure 1. Workflow of the proposed method

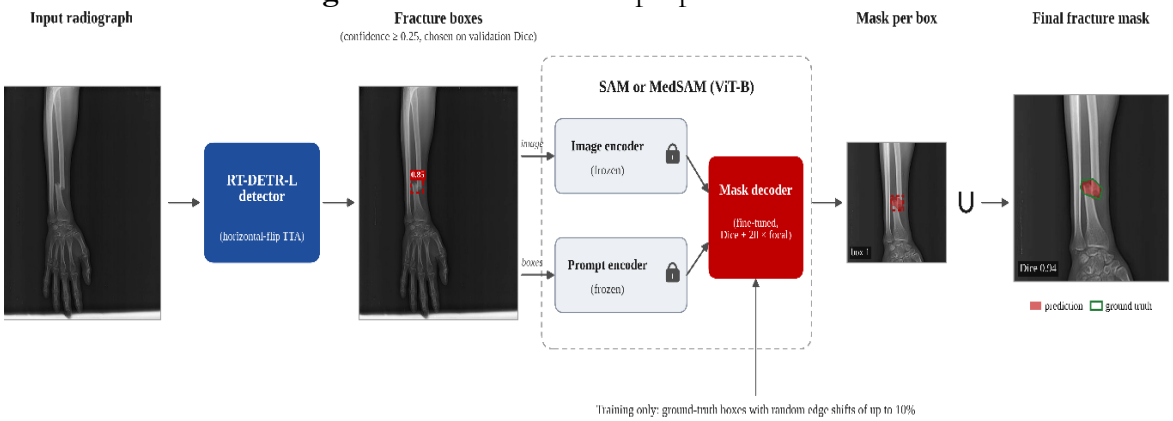

Figure 2. Two-stage pipeline on a test radiograph

Dataset and Split:

FracAtlas [6] contains 4,083 musculoskeletal radiographs of the hand, leg, hip and shoulder. Of these, 717 contain fractures, and each fracture instance is marked with a polygon mask. Figure 3 shows example radiographs with their masks. The polygons were rasterized, and a bounding box around each polygon was used as the ground-truth box. The 717 fractured images were split once, with a fixed random seed, into training (573 images), validation (86 images, 103 fractures) and test (58 images, 70 fractures) sets. All detectors, segmentation models and baselines used this split, every threshold, checkpoint and ensemble was chosen on the validation set, and each method was evaluated once on the test set.

Only the 717 fractured radiographs were used to train and evaluate the segmentation models, for two reasons. First, only these images have fracture masks, so only they provide training targets for a mask decoder. Second, a box-prompted decoder is only called when a box exists. On a normal radiograph a correct pipeline produces no box and therefore no mask, so the behaviour of the pipeline on normal images is decided by the detector and not by the decoder. Adding normal images to decoder training would only add empty targets for boxes that should not exist.

Normal radiographs were therefore used where they matter, in the detector. A random set of 57 normal images, equal to 10% of the fractured training images, was added to detector training as background images with empty labels. Another 300 normal images were never used in training and were kept to measure false alarms at the image level.

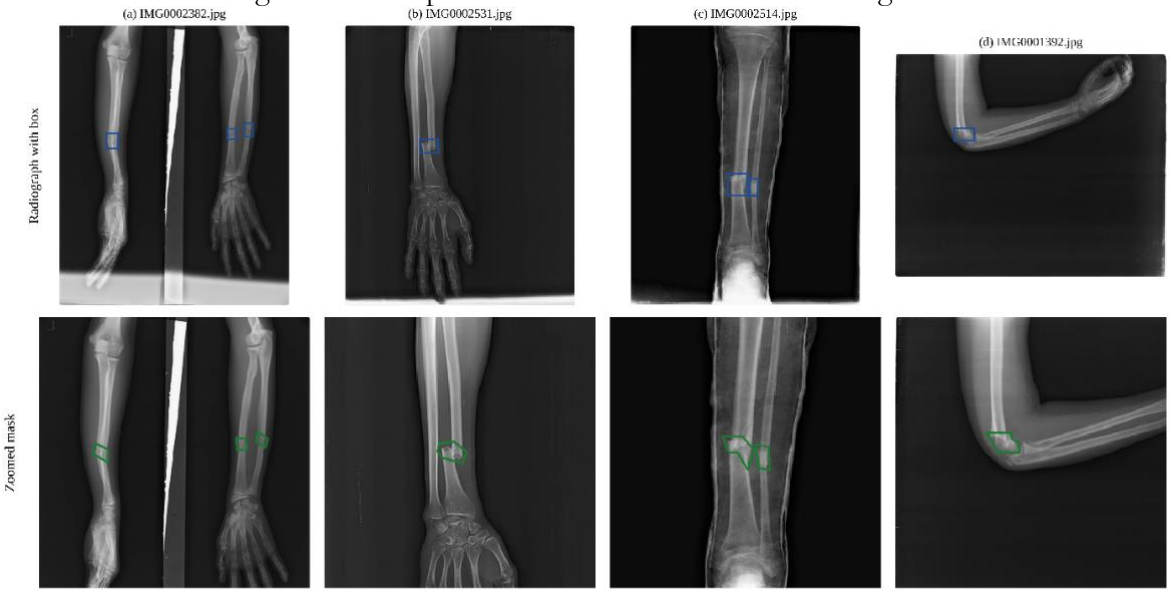

Figure 3. Example radiographs and fracture masks

Fracture regions are small. Across all 922 annotated instances, the median fracture covers 0.26% of the image but 61.3% of its own box (Figure 4). A box prompt therefore turns an extremely imbalanced whole-image problem into a much more balanced problem inside the box.

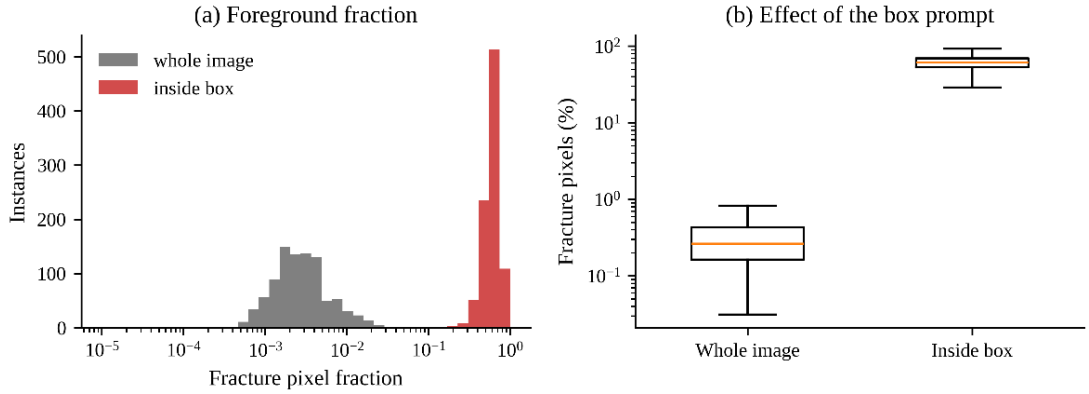

Figure 4. Fracture pixel fraction in the image and in the box

Detection Stage:

The Ultralytics framework [42] was used to train two detectors. YOLO11m was trained at 1024 pixels for 150 epochs using the AdamW optimiser with initial learning rate 0.001 and cosine schedule, a batch size of 8 and an early stopping criterion of 40 epochs. Augmentation involved mosaic, horizontal flips, rotation up to 10 degrees, translation, scaling and brightness changes. Hue and saturation changes were not possible as the images are

greyscale. RT-DETR-L [20] was trained at 1024 resolution, using the same images, for up to 100 epochs with AdamW (learning rate 0.0001), batch size 4 and patience of 30 epochs. Both detectors started from the COCO-pretrained Ultralytics checkpoints yolo11m.pt (about 20 million parameters) and rtdetr-l.pt (about 32 million parameters). YOLO11m ran for all 150 epochs, and RT-DETR-L stopped early after 97 epochs.

Four candidate detectors were created: YOLO11 with and without the Ultralytics test-time augmentation, and RT-DETR with and without horizontal-flip test-time augmentation, where boxes that were predicted on the flipped image were mapped back and kept as a separate member. Weighted Boxes Fusion [40] was applied to each subset of two or more members with fusion IoU of 0.45, 0.55 and 0.65, and two score fusion rules. Single members were also kept as candidates. For each candidate, the confidence threshold was chosen on the validation set to maximise instance F1, where a predicted box counts as correct if its IoU with an unmatched ground-truth box is at least 0.5. Validation F1 was used to select the candidate for segmentation. This F1-based threshold, 0.58 for the selected detector, is used only to report detection results (Table 1 and the image-level metrics) and to choose between candidates. The threshold used to pass boxes to the segmentation models is selected separately (see Detector threshold for segmentation).

Promptable Segmentation Models:

The ViT-B versions of SAM [7] and MedSAM [8] were loaded with the Hugging Face Transformers library [43] from the public checkpoints facebook/sam-vit-base and flaviagiammarino/medsam-vit-base. Each model has about 93.7 million parameters: about 89.7 million in the image encoder, about 6 thousand in the prompt encoder and about 4.06 million (4.3%) in the mask decoder. All radiographs were converted to three channels, resized so that the longer side was 1024 pixels and padded to a square, as in the original models. The image embedding of each picture was computed once and cached. The mask decoder produced a 256 × 256 logit map for each box, which was upsampled to the original image size and thresholded at 0.5. The masks from all boxes of an image were combined by union, and an image without a box received an empty mask.

Fine-tuning of the mask decoder:

The image encoder and the prompt encoder were frozen and only the mask decoder was trained, so about 4.06 million parameters were updated for each model. Each training sample was one fracture instance with its box. The ground-truth masks were resampled to the decoder resolution of 256 × 256 with area averaging, which keeps thin fractures as soft targets. The loss was the sum of the Dice loss [29] and the focal loss [30], with the focal term weighted by 20 as in SAM [7]:

$$\mathcal{L} = \mathcal{L}_{\text{Dice}} + 20\,\mathcal{L}_{\text{focal}} \quad (1)$$

$$\mathcal{L}_{\text{Dice}} = 1 - \frac{2\sum_i p_i g_i + 1}{\sum_i p_i + \sum_i g_i + 1} \quad (2)$$

In Equations (1) to (3), p_i is the predicted probability of pixel i, obtained by applying the sigmoid function to the decoder logit, and g_i is the target value of pixel i in the 256 × 256 target map. The value of g_i lies between 0 and 1 because the area resampling gives fractional values at the fracture border. N = 65,536 is the number of pixels in the map, and the constant 1 in Equation (2) prevents division by zero. The focal loss is

$$\mathcal{L}_{\text{focal}} = \frac{1}{N}\sum_{i=1}^{N} \alpha_i \left(1 - p_{t,i}\right)^{\gamma} \ell_i \quad (3)$$

where $\ell_i = -[g_i \log p_i + (1 - g_i) \log(1 - p_i)]$ is the binary cross-entropy of pixel i, $p_{t,i} = p_i g_i + (1 - p_i)(1 - g_i)$ is the probability given to the target, $\alpha_i = \alpha\, g_i + (1 - \alpha)(1 - g_i)$ is the class weight, $\alpha = 0.25$ and $\gamma = 2$ [30]. The term $(1 - p_{t,i})^{\gamma}$ reduces the loss of pixels that are already classified well, which are mostly background pixels. Both losses are computed for each instance and averaged over the batch.

The weight of 20 follows the 20 to 1 ratio of focal to Dice loss used to train SAM [7]. Because the focal loss is averaged over all 65,536 pixels and down-weights easy background pixels, its value is much smaller than the Dice loss for a small fracture, and the weight keeps the pixel-level term from being ignored. The weight was taken from SAM and was not tuned, so that no additional hyperparameter had to be selected on the small validation set. SAM and MedSAM were trained with exactly the same loss, weights, box jitter, optimiser and schedule.

During training, each edge of the box was shifted at random by up to 10% of the box width or height, so that the decoder learned to tolerate imperfect detector boxes. Horizontally flipped images were added with their own cached embeddings. Training ran for 50 epochs with AdamW (learning rate 0.0001, weight decay 0.01, cosine schedule) and batches of 16 instances. The decoder with the highest validation Dice underground-truth boxes was kept.

Detector Threshold for Segmentation:

Two detector thresholds were used for two different purposes, and both were chosen on the validation set only. The first, the threshold that maximised validation F1 (0.58 for the selected detector), was used to report detection results and to choose between detector candidates (see Detection stage). The second was used to pass boxes to SAM or MedSAM. A threshold that is best for detection F1 is not necessarily best for segmentation, so this threshold was chosen separately for each of the four promptable models: the fully automatic pipeline was run on the 86 validation images for every threshold from 0.05 to 0.90 in steps of 0.05, and the threshold with the highest mean validation Dice was kept. The confidence threshold of the YOLO11m-seg baseline was chosen in the same way. The test images were not used for training or for choosing any checkpoint, threshold, detector or ensemble. They were evaluated once, after all choices had been fixed.

Baseline:

YOLO11m-seg is an end-to-end instance segmentation model. It was initialized from the COCO-pretrained yolo11m-seg.pt checkpoint (about 22 million parameters) and trained on the same images and background images as the YOLO11 detector, at 1024 pixels for up to 150 epochs (early-stopping patience of 40) with the same optimizer, schedule and augmentation. Its prediction was the union of its instance masks at the original resolution, using the confidence threshold with the highest validation Dice on the same grid.

Evaluation:

The Dice coefficient, intersection over union (IoU), pixel precision and recall, and the 95th percentile Hausdorff distance (HD95) in pixels [44] were computed for each test image. An empty prediction on a fractured image scores a Dice of zero. HD95 is defined only when both the prediction and the ground truth are non-empty, so its median is reported over those images. Global Dice was computed by pooling the pixels of all test images. The 95% confidence intervals of the means were computed with 10,000 bootstrap resamples of the test images [45]. The error bars in Figure 6 come from the original analysis with 2,000 resamples and differ from Table 2 by at most 0.004. Paired differences in per-image Dice were tested with the Wilcoxon signed-rank test [46], and the Holm method [47] corrected for the six planned comparisons.

The Wilcoxon signed-rank test does not assume normally distributed differences, but it assumes that the pairs are independent and that the differences are symmetric about their median. Each test image contributes one pair, so the pairs are independent. The Shapiro-Wilk test rejected the normality of the differences in three of the six comparisons ($p < 0.05$), which supports a rank-based test instead of a paired t-test. Images with identical scores, for example when both methods scored a Dice of zero, carry no information about the direction of the difference and are dropped by the test. Because some differences were skewed, the paired bootstrap 95% confidence interval of the mean difference, which does not assume symmetry,

is reported as well, together with the matched-pairs rank-biserial correlation r as the effect size.

Detection was evaluated with instance precision, recall and F1-score at an IoU of 0.5, AP50 and image-level recall, which is the fraction of fractured images in which at least one fracture is detected. The 95% confidence intervals of precision, recall, F1-score and AP50 were computed with 10,000 bootstrap resamples of the test images. Image-level accuracy, sensitivity and specificity were computed from the 58 fractured test images and the 300 held-out normal images.

Robustness to imperfect boxes was evaluated by moving each edge of each ground-truth box at random by up to 5%, 10%, 20% or 30% of the box size, with three repetitions for each level. To evaluate performance by fracture size, the test images were divided into tertiles of fracture size (fracture area divided by image area).

All experiments were run on Kaggle with one NVIDIA T4 GPU (16 GB), Python with PyTorch, Ultralytics 8.4.163 and Transformers 5.0.0, and a fixed random seed of 42. Because the image embeddings were computed once and cached in half precision, each fine-tuning step runs only the mask decoder. The detection notebook, which trained RT-DETR-L and evaluated all detector candidates, ran for about 2.6 hours, and the segmentation notebook, which cached the image embeddings, fine-tuned both decoders, trained the YOLO11m-seg baseline and ran all evaluations, ran for about 4.0 hours.

Results:

Detection:

Of the detector candidates, RT-DETR with horizontal-flip test-time augmentation reached the best validation F1 (0.702) and was selected. The best fused ensemble, which combined all four members, reached 0.700 and was not chosen. Table 1 shows each detector at its own validation-selected threshold. On the test set, the selected detector reached a precision of 0.709 (95% CI 0.607 to 0.812), a recall of 0.557 (0.431 to 0.682), an F1-score of 0.624 (0.509 to 0.729) and an AP50 of 0.579 (0.448 to 0.711). The intervals are wide because the test set contains 70 fractures.

Table 1. Detection results on the test set

Detector	Threshold	Validation F1	Precision	Recall	F1	AP50	Image recall
YOLO11m	0.44	0.663	0.681	0.457	0.547	0.500	0.483
YOLO11m + TTA	0.59	0.678	0.686	0.500	0.579	0.532	0.534
RT-DETR-L	0.54	0.698	0.722	0.557	0.629	0.544	0.586
RT-DETR-L + flip TTA (selected)	0.58	0.702	0.709	0.557	0.624	0.579	0.586
95% CI (selected)			0.607 to 0.812	0.431 to 0.682	0.509 to 0.729	0.448 to 0.711	

Compared with the best YOLO11 configuration, the selected RT-DETR configuration detected four additional fractures among the 70 test fractures and raised test F1 from 0.579 to 0.624 and AP50 from 0.532 to 0.579. Figure 5 shows the precision-recall curves of the detectors. At the image level, the selected detector reached an accuracy of 0.872, a sensitivity of 0.724 (95% CI 0.603 to 0.828), a specificity of 0.900 (0.867 to 0.933) and a balanced accuracy of 0.812 on the 58 fractured test images and 300 held-out normal images. At the lower threshold of 0.25 used to pass boxes to the segmentation models (see Detector threshold for segmentation), 130 of the 300 normal images (43.3%) received at least one box (333 boxes in total), a specificity of 0.567 (0.510 to 0.623). With a looser matching criterion

(IoU 0.3), test recall and F1 rose to 0.657 and 0.736, which means that some of the missed matches are boxes placed on the fracture but with a different size from the annotation. A YOLO11 model trained with region-balanced offline augmentation, in which images of less frequent body regions were copied more often, did not perform better on the validation set (F1 0.663 against 0.678) and was not evaluated further.

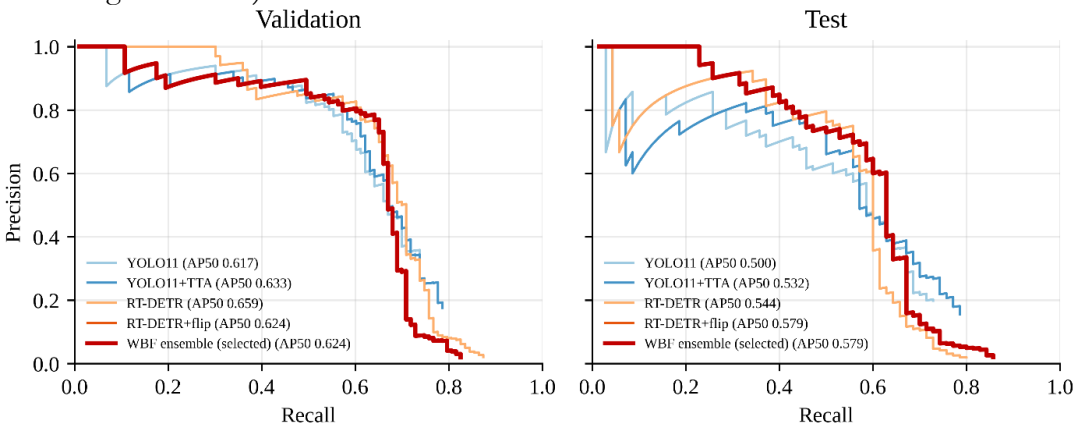

Figure 5. Precision-recall curves of the detectors

Segmentation with ground-truth boxes:

With ground-truth boxes, fine-tuning significantly improved both models (Table 2). Mean Dice rose from 0.701 to 0.843 for SAM and from 0.736 to 0.833 for MedSAM, mean IoU rose from 0.554 (95% CI 0.515 to 0.591) to 0.735 (0.708 to 0.761) for SAM and from 0.592 (0.560 to 0.623) to 0.720 (0.694 to 0.745) for MedSAM, and median HD95 fell from 6.5 to 4.1 pixels and from 6.0 to 5.2 pixels. The zero-shot masks were generally located correctly but underestimated the fracture size (Table 3): zero-shot SAM reached a pixel precision of 0.804 and a recall of 0.667, whereas fine-tuned SAM reached 0.836 and 0.861. Figure 6 shows the mean Dice of all methods with their confidence intervals, and Figure 7 shows the per-image distributions of Dice and HD95.

Table 2. Segmentation results on the test set

Method	Prompt	Mean Dice (95% CI)	Median Dice	IoU (95% CI)	Global Dice	HD95 (px)	Empty
SAM, zero-shot	Ground truth	0.701 (0.667 to 0.733)	0.717	0.554 (0.515 to 0.591)	0.649	6.5	0
MedSAM, zero-shot	Ground truth	0.736 (0.710 to 0.761)	0.751	0.592 (0.560 to 0.623)	0.757	6.0	0
SAM, fine-tuned	Ground truth	0.843 (0.823 to 0.861)	0.866	0.735 (0.708 to 0.761)	0.873	4.1	0
MedSAM, fine-tuned	Ground truth	0.833 (0.814 to 0.851)	0.853	0.720 (0.694 to 0.745)	0.836	5.2	0
SAM, zero-shot	RT-DETR	0.477 (0.399 to 0.551)	0.576	0.359 (0.297 to 0.421)	0.564	16.7	3
MedSAM, zero-shot	RT-DETR	0.521 (0.443 to 0.595)	0.634	0.402 (0.340 to 0.468)	0.609	12.7	3

SAM, fine-tuned	RT-DETR	0.566 (0.481 to 0.646)	0.683	0.459 (0.384 to 0.531)	0.655	12.8	3
MedSAM, fine-tuned	RT-DETR	0.558 (0.472 to 0.641)	0.703	0.450 (0.375 to 0.522)	0.641	12.9	3
YOLO11m-seg	None	0.540 (0.451 to 0.626)	0.686	0.438 (0.360 to 0.514)	0.559	17.2	5

Table 3. Pixel precision and recall on the test set

Method	Prompt	Pixel precision (95% CI)	Pixel recall (95% CI)
SAM, zero-shot	Ground truth	0.804 (0.759 to 0.845)	0.667 (0.621 to 0.709)
MedSAM, zero-shot	Ground truth	0.840 (0.800 to 0.878)	0.676 (0.646 to 0.708)
SAM, fine-tuned	Ground truth	0.836 (0.805 to 0.866)	0.861 (0.845 to 0.876)
MedSAM, fine-tuned	Ground truth	0.809 (0.778 to 0.839)	0.874 (0.856 to 0.891)
SAM, zero-shot	RT-DETR	0.521 (0.429 to 0.611)	0.497 (0.413 to 0.578)
MedSAM, zero-shot	RT-DETR	0.576 (0.483 to 0.665)	0.527 (0.446 to 0.604)
SAM, fine-tuned	RT-DETR	0.539 (0.451 to 0.625)	0.648 (0.549 to 0.737)
MedSAM, fine-tuned	RT-DETR	0.521 (0.434 to 0.605)	0.660 (0.563 to 0.753)
YOLO11m-seg	None	0.518 (0.428 to 0.607)	0.614 (0.518 to 0.704)

Zero-shot MedSAM was more accurate than zero-shot SAM, but this ranking did not persist after fine-tuning. Table 4 gives the paired tests. The fine-tuning gains were large and consistent: fine-tuning improved 54 of the 58 test images for SAM and 53 for MedSAM, with effect sizes of r = 0.98 and 0.95. With RT-DETR boxes the gain from fine-tuning MedSAM was smaller but still significant (+0.037 Dice, 95% CI 0.015 to 0.061, r = 0.49). The three comparisons that were not significant should be read in light of their confidence intervals. Fine-tuned MedSAM and SAM differed by -0.010 Dice (95% CI -0.021 to 0.002) with ground-truth boxes and by -0.007 (-0.017 to 0.003) with RT-DETR boxes, suggesting that any difference between the two fine-tuned models is likely to be small in practice. For the automatic pipeline against YOLO11m-seg, the interval was wide (-0.044 to 0.081) and the effect size close to zero (r = -0.01), so the 58 test images cannot show that either method is better, and the two are reported as comparable, not as equivalent. When the same models were retrained in four separate runs during this study, the Dice with ground-truth boxes ranged only from 0.841 to 0.846 for SAM and from 0.833 to 0.834 for MedSAM.

Table 4. Paired significance tests on test Dice

Comparison	Mean Dice A	Mean Dice B	Difference (95% CI)	Effect size r	p (Holm)
SAM fine-tuned vs zero-shot, ground-truth box	0.843	0.701	+0.142 (0.112 to 0.175)	0.98	< 0.001
MedSAM fine-tuned vs zero-shot, ground-truth box	0.833	0.736	+0.097 (0.077 to 0.117)	0.95	< 0.001
MedSAM fine-tuned vs zero-shot, RT-DETR box	0.558	0.521	+0.037 (0.015 to 0.061)	0.49	0.013
MedSAM vs SAM, both fine-tuned, ground-truth box	0.833	0.843	-0.010 (-0.021 to 0.002)	-0.26	0.263

MedSAM vs SAM, both fine-tuned, RT-DETR box	0.558	0.566	-0.007 (-0.017 to 0.003)	-0.23	0.358
MedSAM fine-tuned, RT-DETR box vs YOLO11m-seg	0.558	0.540	+0.019 (-0.044 to 0.081)	-0.01	0.973

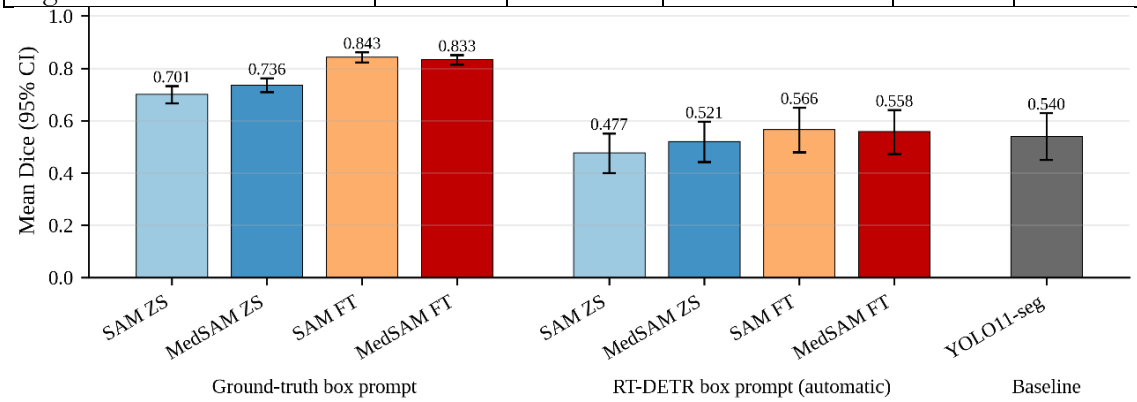

Figure 6. Mean test Dice with 95% confidence intervals

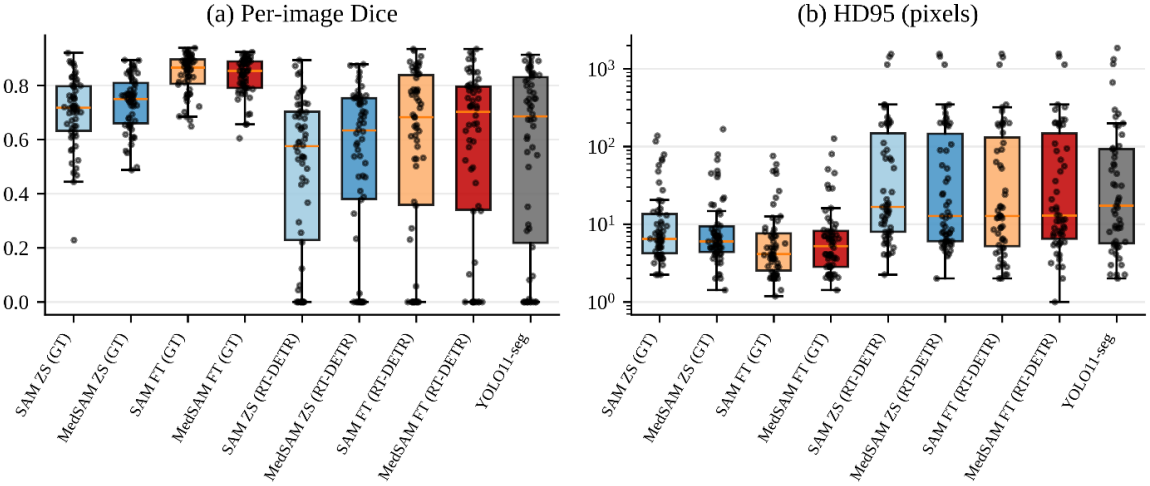

Figure 7. Per-image Dice and HD95

Automatic pipeline:

With RT-DETR boxes, fine-tuning still improved MedSAM significantly, from 0.521 to 0.558 mean Dice. The detector threshold chosen on validation Dice was 0.25 for all four promptable models, well below the threshold of 0.58 that maximised detection F1 (Figure 8).

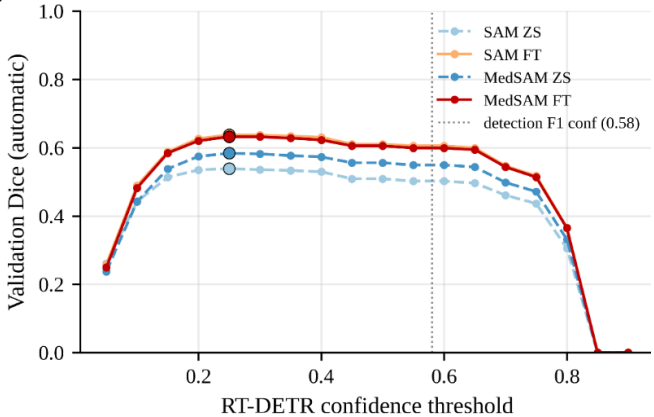

Figure 8. Validation Dice against detector threshold

Table 5 separates the effect of the detector from the effect of the threshold rule for fine-tuned MedSAM. Each row comes from a separate run in which the decoder was fine-

tuned again, and the ground-truth-box results across these runs differed by no more than 0.005 Dice. The value of 0.465 in the first row is the usual way of connecting a detector to a segmentation model: the YOLO11 detector at its detection-F1 threshold. The value of 0.558 in the last row is the configuration reported in Table 2: the RT-DETR detector at the threshold chosen on validation Dice.

Table 5. Effect of detector and threshold rule

Detector	Threshold rule	Threshold	Mean Dice	Empty
YOLO11m + TTA	Detection F1	0.59	0.465	17
RT-DETR-L + flip TTA	Detection F1	0.58	0.499	16
YOLO11m + TTA	Validation Dice	0.20	0.512	10
RT-DETR-L + flip TTA	Validation Dice	0.25	0.558	3

Both changes helped, and their effects were approximately additive. Changing only the detector from YOLO11 to RT-DETR added 0.034 Dice under the F1 rule (0.465 to 0.499) and 0.046 under the Dice rule (0.512 to 0.558). Changing only the threshold rule added 0.047 Dice with YOLO11 (0.465 to 0.512) and 0.059 with RT-DETR (0.499 to 0.558). The total gain of 0.093 is slightly larger than the sum of the two single changes (0.081), which indicates a small positive interaction of 0.012. Most of the improvement came from the reduction in empty predictions, which fell from 17 to 3 test images. With RT-DETR boxes at the selected threshold, 11 of the 58 test images still scored a Dice of zero for each promptable model: 3 had an empty prediction and 8 had a prediction that did not overlap the annotated fracture. For YOLO11m-seg, 11 images also scored zero, 5 of them with an empty prediction.

Against the YOLO11m-seg baseline, the automatic pipeline with fine-tuned MedSAM had a higher mean Dice (0.558 against 0.540), median Dice (0.703 against 0.686) and global Dice (0.641 against 0.559), a lower median HD95 (12.9 against 17.2 pixels) and fewer empty predictions (3 against 5). The difference in mean Dice was not significant ($p = 0.97$, 95% CI of the mean difference -0.044 to 0.081), so the two approaches are comparable in overlap.

Robustness to Imperfect Boxes:

Figure 9 shows mean Dice when every edge of the ground-truth box is shifted at random by up to 30% of the box size. Dice scores gradually decreased for all four models as the shift increased. Up to a 10% shift, which matches the shift used during training, fine-tuned SAM and MedSAM lost at most 0.02 Dice (from 0.843 to 0.823 and from 0.833 to 0.815). At a 30% shift, fine-tuned SAM still reached 0.701 and fine-tuned MedSAM 0.697, against 0.612 and 0.648 for the zero-shot models. The fine-tuned models lost more Dice in absolute terms, but they remained more accurate than the zero-shot models at every shift level.

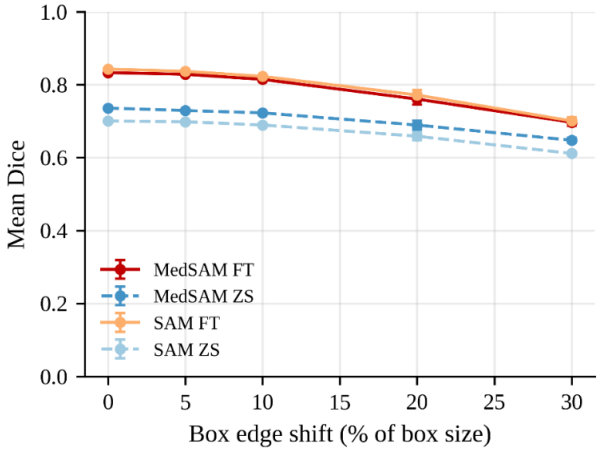

Figure 9. Robustness to shifted boxes

Fracture Size:

Figure 10 reports mean Dice for small, medium and large fractures, defined by tertiles of the fracture area relative to the image area. Each tertile holds 19 or 20 test images. With ground-truth boxes, fine-tuned MedSAM was almost equally accurate for small, medium and large fractures (0.813, 0.842 and 0.844), so the decoder itself handles small fractures well. With RT-DETR boxes, its Dice fell to 0.321 for small fractures, against 0.643 for medium and 0.704 for large ones, and YOLO11m-seg showed the same pattern (0.342, 0.596 and 0.674). The automatic pipeline was more accurate than YOLO11m-seg for medium and large fractures and slightly less accurate for small ones. Because the same decoder performs well on small fractures when given ground-truth boxes, the low Dice scores with RT-DETR boxes suggest that detector-box quality is a major contributor to the reduced performance on small fractures.

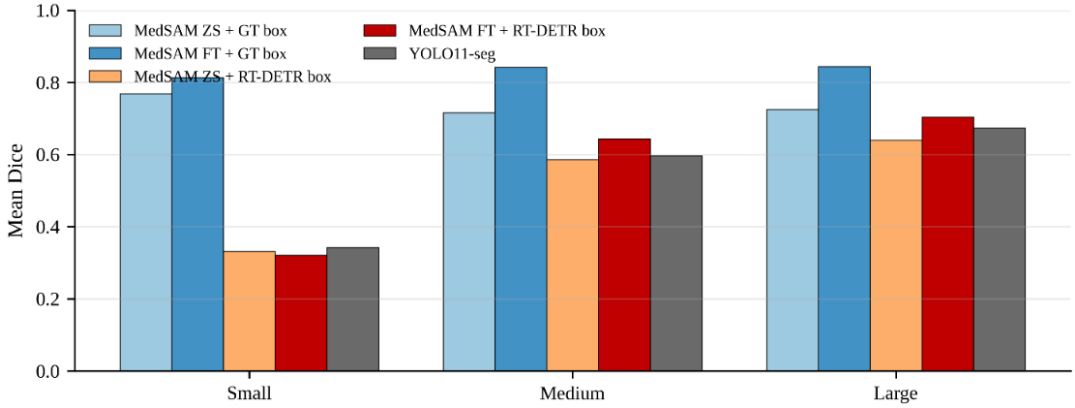

Figure 10. Test Dice by fracture size

Qualitative Results:

Figure 11 compares the methods on five test images chosen at fixed quantiles of the automatic pipeline's Dice. Figure 12 shows the four test images with the lowest Dice for the automatic pipeline, together with the RT-DETR boxes.

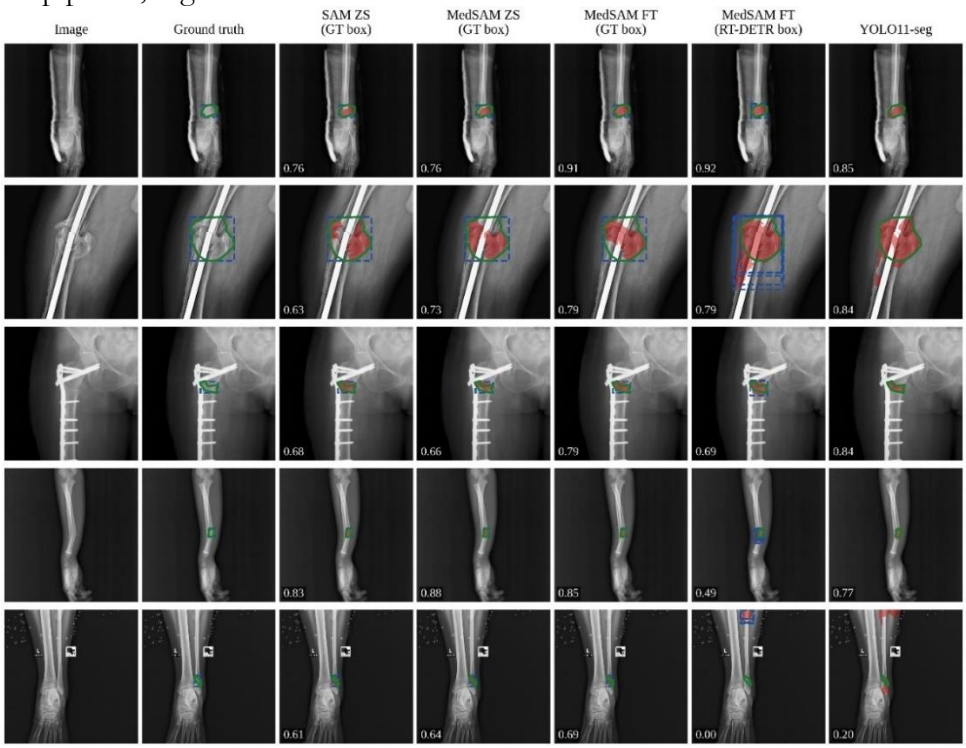

Figure 11. Qualitative comparison of the methods

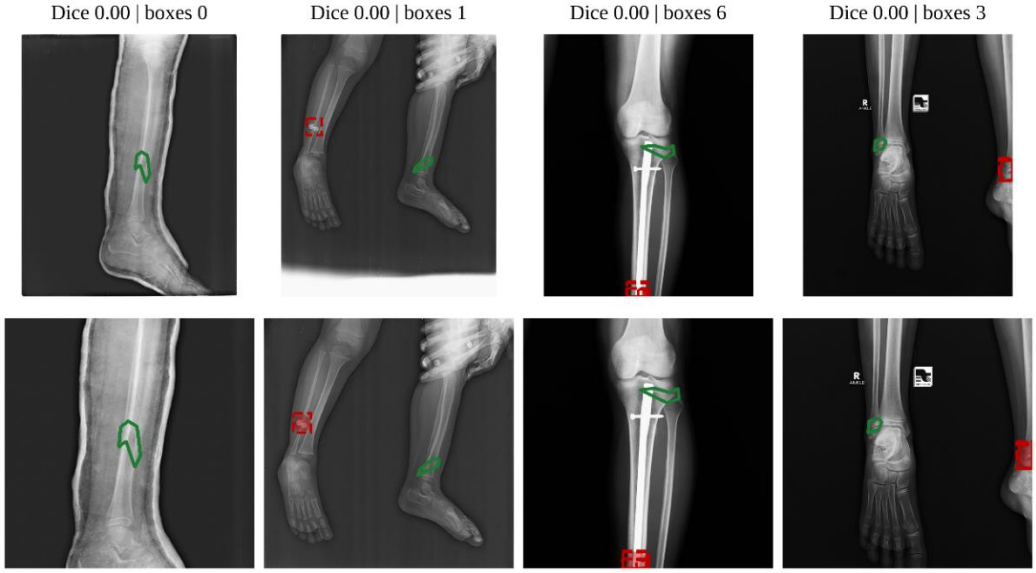

Figure 12. Failure cases of the automatic pipeline

Discussion:

With a suitable box, a SAM or MedSAM decoder segments the fractures in FracAtlas with a mean Dice of about 0.84. Only the mask decoder was trained, and the image encoder was not changed. The zero-shot models generally placed their masks in the correct area but tended to cover too little of the fracture, while fine-tuning mainly increased pixel recall. This suggests that the pretrained encoders already represent the structure of the radiograph well, and that the decoder only has to learn how much of the region belongs to the fracture.

Medical pretraining helped only without fine-tuning. Zero-shot MedSAM was 0.035 Dice better than zero-shot SAM, but after the decoder was fine-tuned on 573 images the two models did not differ, with either prompt. When a few hundred labelled images are available, the general SAM encoder is therefore sufficient for fracture segmentation.

The automatic pipeline shows that the detector threshold should be chosen for the task that follows detection. Detection F1 penalizes missed fractures and false-positive boxes, whereas segmentation evaluation can assign a Dice score of zero when no mask is predicted for a fractured image. A false box, by contrast, may add only a small false-positive region. Choosing the threshold on validation Dice lowered it from 0.58 to 0.25 and reduced the number of empty predictions with RT-DETR boxes from 16 to 3. This choice costs nothing and should be reported whenever a detector is connected to a promptable segmentation model. The threshold was, however, chosen on fractured validation images only, so it does not account for false alarms on normal radiographs: at 0.25, 43% of the held-out normal images received at least one box, against 10% at 0.58. The Dice-based threshold therefore suits outlining fractures in radiographs that are known or suspected to be fractured. For unselected radiographs, the higher detection threshold or a separate image-level classifier should decide whether any box is passed to the segmentation model.

An additional gain was provided by the transformer detector. RT-DETR found more fractures than YOLO11 at a similar precision and its detections led to higher segmentation Dice under both threshold rules. Fusing YOLO11 with RT-DETR did not improve validation F1, which suggests that most fractures found by YOLO11 were also found by RT-DETR.

The automatic pipeline was comparable to an end-to-end YOLO11m-seg model in mean Dice and better in boundary distance and global Dice. Its main advantage is that detection and segmentation are separated: the same decoder reached a Dice of 0.833 with ground-truth boxes but 0.558 with detector boxes. The remaining error therefore comes

mainly from the detector, and any improvement in detection recall will directly improve the segmentation, whereas an end-to-end model does not separate the two sources of error.

Comparison with Existing Studies:

Table 6 places the results of this study next to published results on FracAtlas and on another radiographic fracture segmentation task. The studies differ in three ways that limit direct comparison. They use different data (all of FracAtlas, a random split of it, or a different dataset of a single body region), different evaluation units (per image or per instance) and different metrics (box AP50, mask mAP50, Dice or IoU). Detection AP50 and mask mAP50 are ranking metrics over all instances and confidence levels, whereas Dice and IoU measure pixel overlap at one operating point, so they cannot be converted into each other. The last column of Table 6 states for each study what can and cannot be compared. Only values of the same metric are compared below, and none of them uses the split of this study.

Table 6. Comparison with published results

Study	Method	Data and split	Task	Reported result	Comparable with this study
[6]	YOLOv8s	FracAtlas, random 80/12/8, validation set	Detection	mAP50 0.562, P 0.807, R 0.473	Partly: same metric, different split
[6]	YOLOv8s-seg	FracAtlas, random 80/12/8, validation set	Segmentation	Mask mAP50 0.589, P 0.830, R 0.499	No: mask mAP50, not Dice
[26]	WCAY (YOLOv8s based)	FracAtlas, random 70/20/10	Detection	mAP50 0.567	Partly: same metric, different split
[27]	YOLOv7-ATT	FracAtlas, generalisation experiment	Detection	mAP 0.862	No: different protocol
[31]	DeepLabV3 (ResNet)	FracAtlas, own split	Segmentation	IoU above 0.93	No: different split and averaging
[32]	YOLOv8	Hip radiographs, single region	Segmentation	Dice 0.77	No: different dataset (hip only)
[16]	Zero-shot MedSAM, ground-truth box	FracAtlas, all fractured images, per instance	Segmentation	Mean IoU 0.249 (dataset box), 0.317 (tight box)	Partly: same model, per-instance scoring, different boxes
[16]	YOLOv8x-seg, MedSAM-refined labels	FracAtlas, 5-fold CV and held-out test set	Segmentation	Mask mAP50 0.63 (test)	No: mask mAP50, not Dice

This study	RT-DETR-L + flip TTA	FracAtlas, seeded 80/12/8	Detection	AP50 0.624 (val), 0.579 (test)	Reference
This study	YOLO11m-seg	FracAtlas, seeded 80/12/8	Segmentation	Dice 0.540	Reference
This study	RT-DETR + fine-tuned MedSAM	FracAtlas, seeded 80/12/8	Segmentation	Dice 0.558	Reference
This study	Zero-shot MedSAM, ground-truth box	FracAtlas, seeded 80/12/8, per image	Segmentation	Dice 0.736, IoU 0.592	Reference
This study	Fine-tuned SAM, ground-truth box	FracAtlas, seeded 80/12/8	Segmentation	Dice 0.843	Reference

For detection, the selected RT-DETR configuration reached an AP50 of 0.624 on the validation set and 0.579 on the test set. The dataset paper reported 0.562 for YOLOv8s on its validation set [6], and the WCAY detector reached 0.567 on a 70/20/10 split [26]. These values use the same metric but different splits, and the 95% confidence interval of the test AP50 of this study (0.448 to 0.711) contains both of them, so no claim of better detection is made. The mAP of 0.862 reported for YOLOv7-ATT [27] comes from a generalisation experiment with a different protocol and is not comparable.

For segmentation, the dataset paper [6] and [16] reported mask mAP50 values (0.589 and 0.63), which cannot be converted into Dice. [31] reported an IoU above 0.93 with DeepLabV3, but their split, preprocessing and averaging of the metric differ, so this value is not comparable with the per-image IoU of this study (0.450 for the automatic pipeline and 0.735 for fine-tuned SAM with ground-truth boxes). [32] reported a Dice of 0.77 for hip fractures with YOLOv8, on a large dataset of a single body region. This is the same metric but a different task and dataset. The comparison therefore shows where this study stands in the literature, not which method is better.

[16] reported a much lower zero-shot MedSAM overlap on FracAtlas, a mean IoU of 0.249 to 0.317, than the zero-shot MedSAM result of this study, a mean IoU of 0.592 with ground-truth boxes (Table 2). The protocols differ in three ways. First, they scored each fracture instance separately, whereas this study scored the union mask of each image. Second, their prompts were the boxes supplied with the dataset or polygon boxes padded by 4 pixels, whereas this study uses tight polygon boxes. Third, they evaluated all 717 fractured images, whereas this study reports the 58 test images. A direct comparison of the two values is therefore not reliable, but both studies found that zero-shot masks miss part of the fracture extent. Their final YOLOv8x-seg model reached a mask mAP50 of 0.63 without any box prompt, a metric not computed in this study. They did not fine-tune MedSAM or prompt it with detector boxes, which is the setting studied here.

Clinical and Practical Implications:

A fracture mask adds information that a box or a label does not give. The area, length and shape of the fracture can be measured from it, and the same measurement can be repeated

at follow-up to judge healing or displacement. Automatic segmentation of bones on radiographs of distal radius fractures already allows radiologic angles and heights to be measured with close agreement to manual measurement [4], and fracture masks could support similar measurements of the fracture itself. The box-prompted design suits two modes of use. In an interactive mode, a radiologist or orthopaedic surgeon draws a box around a fracture and receives its outline, which corresponds to the ground-truth-box setting of this study (mean Dice 0.84). In an automatic mode, the detector proposes the boxes, which could serve as a second reader that marks and outlines suspected fractures for review or prioritises studies in a busy emergency department. The same interactive tool could also reduce the cost of producing pixel labels for new datasets.

The results also show the limits of the current pipeline for clinical practice. The detector found 56% of the fracture instances at its F1 threshold and flagged 72% of the fractured test images, so the system cannot be used to rule out a fracture. It placed at least one false box on 10% of the held-out normal radiographs at that threshold, and on 43% of them at the lower segmentation threshold of 0.25, which would create false masks that a reader has to dismiss. Measurements are in pixels, because FracAtlas does not provide reliable pixel spacing, so physical sizes would require calibration. The model was developed on one public dataset and has not been tested on other hospitals, devices or patient groups, or in a reader study. For these reasons the pipeline is a decision-support tool under human supervision, not a stand-alone diagnostic system.

Limitations:

Only one dataset and one split were used. The split is not the official FracAtlas split, so the numbers are not directly comparable with results reported on other splits. The test set comprises 58 images and 70 fractures, which gives wide confidence intervals for the automatic pipeline and limits the power of the paired tests. Only the ViT-B versions of the models were tested and only the mask decoder was fine-tuned. Adapting the image encoder, for example with low-rank adapters, may give further gains. The validation scores are somewhat optimistic, because the thresholds and the detector were selected on 86 validation images, but the test set was not used for any selection. The segmentation models were trained only on fractured radiographs, so their behaviour on normal radiographs depends on the false boxes of the detector, which are frequent at the segmentation threshold (43% of normal images at 0.25). The median HD95 was computed over images with a non-empty prediction and is given in pixels, because the radiographs do not contain reliable pixel spacing. Finally, the FracAtlas masks are coarse polygons, so part of the remaining error comes from the annotation style rather than from the model.

Recommendations and Future Research:

Five directions follow from these results. First, the pipeline should be validated externally on radiographs from other hospitals and devices, ideally in a multicentre study with a pre-specified protocol, and in a reader study that measures whether the masks change clinical decisions. Second, larger training sets are needed, and the box-prompted model itself could be used to speed up the annotation of new fracture masks. Third, the approach should be tested on other anatomical regions, such as the wrist, ribs, spine and pelvis, and on other modalities such as CT, where fracture segmentation has also been studied [5][33]. Fourth, the pipeline should be compared with other advanced segmentation models, including nnU-Net [34], adapter-based and low-rank adaptations of SAM [14][37] and newer promptable models such as SAM 2 [38], ideally with repeated splits or cross-validation to obtain more stable estimates. Fifth, because detection recall is the main remaining limit, detectors designed for small objects and joint training of the detector and the decoder should be studied, together with a threshold rule that also controls false alarms on normal radiographs and calibration of pixel spacing so that fracture size can be reported in millimetres.

Conclusion:

After fine-tuning their mask decoders, box-prompted foundation models segment radiographic fractures accurately. Decoder fine-tuning raised mean Dice with ground-truth boxes to 0.843 for SAM and 0.833 for MedSAM, and the advantage of medical pretraining disappeared after fine-tuning on FracAtlas. An RT-DETR detector with a threshold chosen on validation Dice raised the automatic mean Dice from 0.465 to 0.558, comparable with an end-to-end YOLO11m-seg model but with better boundary accuracy. The gap between automatic and ground-truth boxes shows that detection recall is the current limit. Future work should focus on external and multicentre validation, other anatomical regions, comparison with other segmentation models and better detection of small fractures (see Recommendations and Future Research).

Acknowledgement:

The authors thank the creators of the FracAtlas dataset for making the radiographs and annotations publicly available, and Kaggle for providing the GPU resources used for the experiments.

Author's Contribution:

Muhammad Ilyas: conceptualisation, methodology, data preparation, model training and experiments, formal analysis and writing of the original draft. Muhammad Javed Iqbal: supervision, conceptualisation, validation of the study design and results, project administration, and review and editing of the manuscript. Moavia Hassan: software implementation of the detection and segmentation pipelines, visualisation and figure preparation, statistical analysis, and review and editing of the manuscript. Syed Mushhad M. Gilani: literature review, validation of the comparison with existing studies, critical review of the results and discussion, and review and editing of the manuscript. All authors have read and agreed to the final version of the manuscript.

Data Availability:

The FracAtlas dataset is publicly available [6]. The code used in this study is available from the corresponding author on reasonable request.

Conflict of Interest:

The authors declare no conflict of interest.

Declaration:

This manuscript is original, has not been published before and is not under consideration for publication in any other journal. All authors contributed significantly to this work and agree with the content of the manuscript.

Project Details:

This research was not conducted under a funded project.

References:

[1] R. Lindsey, A. Daluiski, S. Chopra, A. Lachapelle, M. Mozer, S. Sicular, D. Hanel, M. Gardner, A. Gupta, R. Hotchkiss, and H. Potter, "Deep neural network improves fracture detection by clinicians," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 115, no. 45, pp. 11591–11596, Nov. 2018, doi: 10.1073/PNAS.1806905115.

[2] M. Kutbi, "Artificial Intelligence-Based Applications for Bone Fracture Detection Using Medical Images: A Systematic Review," *Diagnostics (Basel, Switzerland)*, vol. 14, no. 17, p. 1879, Sep. 2024, doi: 10.3390/DIAGNOSTICS14171879.

[3] R. Y. L. Kuo, C. Harrison, T.-A. Curran, B. Jones, A. Freethy, D. Cussons, M. Stewart, G. S. Collins, and D. Furniss, "Artificial Intelligence in Fracture Detection: A Systematic Review and Meta-Analysis," *Radiology*, vol. 304, no. 1, pp. 50–62, Jul. 2022, doi: 10.1148/RADIOL.211785.

[4] S. Lee, K. G. Kim, Y. J. Kim, J. S. Jeon, G. P. Lee, K.-C. Kim, and S. H. Jeon, "Automatic Segmentation and Radiologic Measurement of Distal Radius Fractures

Using Deep Learning," *Clin. Orthop. Surg.*, vol. 16, no. 1, pp. 113–124, Feb. 2024, doi: 10.4055/CIOS23130.

[5] X. Wang and Y. Wang, "Composite Attention Residual U-Net for Rib Fracture Detection," *Entropy*, vol. 25, no. 3, p. 466, Mar. 2023, doi: 10.3390/E25030466.

[6] I. Abedeen, M. A. Rahman, F. Z. Prottyasha, T. Ahmed, T. M. Chowdhury, and S. Shatabda, "FracAtlas: A Dataset for Fracture Classification, Localization and Segmentation of Musculoskeletal Radiographs," *Sci. Data*, vol. 10, no. 1, pp. 521-, Aug. 2023, doi: 10.1038/s41597-023-02432-4.

[7] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment Anything," *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 3992–4003, Apr. 2023, doi: 10.1109/ICCV51070.2023.00371.

[8] J. Ma, Y. He, F. Li, L. Han, C. You, and B. Wang, "Segment Anything in Medical Images," *Nat. Commun.*, vol. 15, no. 1, p. 654, Jan. 2024, doi: 10.1038/s41467-024-44824-z.

[9] M. A. Mazurowski, H. Dong, H. Gu, J. Yang, N. Konz, and Y. Zhang, "Segment anything model for medical image analysis: An experimental study," *Med. Image Anal.*, vol. 89, p. 102918, Oct. 2023, doi: 10.1016/J.MEDIA.2023.102918.

[10] Y. Huang, X. Yang, L. Liu, H. Zhou, A. Chang, X. Zhou, R. Chen, J. Yu, J. Chen, C. Chen, S. Liu, H. Chi, X. Hu, K. Yue, L. Li, V. Grau, D.-P. Fan, F. Dong, and D. Ni, "Segment anything model for medical images?," *Med. Image Anal.*, vol. 92, p. 103061, Feb. 2024, doi: 10.1016/J.MEDIA.2023.103061.

[11] Y. Zhang, Z. Shen, and R. Jiao, "Segment anything model for medical image segmentation: Current applications and future directions," *Comput. Biol. Med.*, vol. 171, p. 108238, Mar. 2024, doi: 10.1016/J.COMPBIOMED.2024.108238.

[12] H. Gu, H. Dong, J. Yang, and M. A. Mazurowski, "How to build the best medical image segmentation algorithm using foundation models: a comprehensive empirical study with Segment Anything Model," *Mach. Learn. Biomed. Imaging*, vol. 2025, no. May 2025, pp. 88–120, May 2025, doi: 10.59275/j.melba.2025-86a6.

[13] C. Magg, H. Kervadec, and C. I. Sánchez, "Zero-shot capability of SAM-family models for bone segmentation in CT scans," Nov. 2024, Accessed: Nov. 10, 2026. [Online]. Available: https://arxiv.org/pdf/2411.08629

[14] J. Wu, W. Ji, Y. Liu, H. Fu, M. Xu, Y. Xu, and Y. Jin, "Medical sam adapter: Adapting segment anything model for medical image segmentation," *arXiv Prepr. arXiv2304.12620*, vol. 1, 2023, doi: https://doi.org/10.48550/arXiv.2304.12620.

[15] S. Pandey, K.-F. Chen, and E. B. Dam, "Comprehensive Multimodal Segmentation in Medical Imaging: Combining YOLOv8 with SAM and HQ-SAM Models," Oct. 2023, Accessed: Nov. 10, 2026. [Online]. Available: https://arxiv.org/pdf/2310.12995

[16] A. Ahmed, A. S. Imran, Z. Kastrati, and S. M. Daudpota, "Beyond prompt-only segmentation: Foundation model–guided trust-region label refinement for single-stage bone fracture instance segmentation," *Eng. Appl. Artif. Intell.*, vol. 182, p. 115849, Oct. 2026, doi: 10.1016/J.ENGAPPAI.2026.115849.

[17] M. Ilyas, "Bone fracture detection and segmentation using deep learning," unpublished M.S. thesis, Department of Computer Science, University of Engineering and Technology, Taxila, Pakistan, 2026.

[18] U. Muhammad, W. Wang, A. Hadid, and S. Pervez, "Bag of words KAZE (BoWK) with two-step classification for high-resolution remote sensing images," *IET Comput. Vis.*, vol. 13, no. 4, pp. 395–403, Jun. 2019, doi: 10.1049/IET-CVI.2018.5069.

[19] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-End Object Detection with Transformers," *Lect. Notes Comput. Sci. (including Subser.*

Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 12346 LNCS, pp. 213–229, 2020, doi: 10.1007/978-3-030-58452-8_13.

[20] Y. Zhao, W. Lv, S. Xu, J. Wei, G. Wang, Q. Dang, Y. Liu, and J. Chen, "DETRs Beat YOLOs on Real-time Object Detection," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 16965–16974, Jun. 2024, doi: 10.1109/CVPR52733.2024.01605.

[21] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-December, pp. 779–788, Jun. 2016, doi: 10.1109/CVPR.2016.91.

[22] Y. L. Thian, Y. Li, P. Jagmohan, D. Sia, V. E. Yao Chan, and R. T. Tan, "Convolutional Neural Networks for Automated Fracture Detection and Localization on Wrist Radiographs," *Radiol. Artif. Intell.*, vol. 1, no. 1, p. e180001, Jan. 2019, doi: 10.1148/RYAI.2019180001.

[23] E. Nagy, M. Janisch, F. Hržić, E. Sorantin, and S. Tschauner, "A pediatric wrist trauma X-ray dataset (GRAZPEDWRI-DX) for machine learning," *Sci. Data*, vol. 9, no. 1, p. 222, May 2022, doi: 10.1038/S41597-022-01328-Z.

[24] R. Y. Ju and W. Cai, "Fracture detection in pediatric wrist trauma X-ray images using YOLOv8 algorithm," *Sci. Rep.*, vol. 13, no. 1, p. 20077, Nov. 2023, doi: 10.1038/S41598-023-47460-7.

[25] D. Bhattacharya, S. Das, T. Biswas, and G. Mandal, "ResNet based backbone integrated YOLO framework for bone fracture detection," *Sci. Rep.*, vol. 16, no. 1, pp. 12954-, Mar. 2026, doi: 10.1038/s41598-026-41782-y.

[26] P. Chen, S. Liu, W. Lu, F. Lu, and B. Ding, "WCAY object detection of fractures for X-ray images of multiple sites," *Sci. Rep.*, vol. 14, no. 1, pp. 26702-, Nov. 2024, doi: 10.1038/s41598-024-77878-6.

[27] J. Zou and M. R. Arshad, "Detection of whole body bone fractures based on improved YOLOv7," *Biomed. Signal Process. Control*, vol. 91, p. 105995, May 2024, doi: 10.1016/J.BSPC.2024.105995.

[28] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," Lect. Notes Comput. Sci., vol. 9351, pp. 234–241, Oct. 2015, doi: 10.1007/978-3-319-24574-4_28.

[29] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation," *arXiv:1606.04797*, 2016, doi: https://doi.org/10.48550/arXiv.1606.04797.

[30] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020, doi: 10.1109/TPAMI.2018.2858826.

[31] S. Turk, O. Bingol, A. Coskuncay, and T. Aydin, "The impact of implementing backbone architectures on fracture segmentation in X-ray images," *Eng. Sci. Technol. an Int. J.*, vol. 59, p. 101883, Nov. 2024, doi: 10.1016/J.JESTCH.2024.101883.

[32] G. González, J. Galant, J. M. Salinas, E. Benítez, M. D. Sánchez-Valverde, J. Calbo, and N. Cerrolaza, "Classification and segmentation of hip fractures in x-rays: highlighting fracture regions for interpretable diagnosis," *Insights Imaging*, vol. 16, no. 1, pp. 86-, Apr. 2025, doi: 10.1186/S13244-025-01958-Y.

[33] Y. Liu, S. Yibulayimu, G. Zhu, C. Shi, C. Liang, C. Zhao, X. Wu, Y. Sang, and Y. Wang, "Automatic pelvic fracture segmentation: a deep learning approach and benchmark dataset," *Front. Med.*, vol. 12, p. 1511487, Apr. 2025, doi: 10.3389/FMED.2025.1511487.

[34] F. Isensee, P. F. Jaeger, S. A. A. Kohl, J. Petersen, and K. H. Maier-Hein, "nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation," *Nat. Methods*, vol. 18, no. 2, pp. 203–211, Dec. 2020, doi: 10.1038/s41592-020-01008-

z.

[35] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," *ICLR 2021 - 9th Int. Conf. Learn. Represent.*, Oct. 2020, Accessed: Feb. 21, 2024. [Online]. Available: https://arxiv.org/abs/2010.11929v2

[36] K. He, X. Chen, S. Xie, Y. Li, P. Dollar, and R. Girshick, "Masked Autoencoders Are Scalable Vision Learners," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2022-June, pp. 15979–15988, 2022, doi: 10.1109/CVPR52688.2022.01553.

[37] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-Rank Adaptation of Large Language Models," *arXiv:2106.09685*, 2021, [Online]. Available: https://arxiv.org/abs/2106.09685

[38] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K. V. Alwala, N. Carion, C.-Y. Wu, R. Girshick, P. Dollár, and C. Feichtenhofer, "Sam 2: Segment anything in images and videos," *arXiv Prepr. arXiv2408.00714*, 2024, doi: https://doi.org/10.48550/arXiv.2408.00714.

[39] C. Chang, H. Law, C. Poon, S. Yen, K. Lall, A. Jamshidi, V. Malis, D. Hwang, and W. C. Bae, "Segment Anything Model (SAM) and Medical SAM (MedSAM) for Lumbar Spine MRI," *Sensors*, vol. 25, no. 12, p. 3596, Jun. 2025, doi: 10.3390/S25123596.

[40] R. Solovyev, W. Wang, and T. Gabruseva, "Weighted boxes fusion: Ensembling boxes from different object detection models," *Image Vis. Comput.*, vol. 107, p. 104117, Mar. 2021, doi: 10.1016/J.IMAVIS.2021.104117.

[41] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis, "Soft-NMS -- Improving Object Detection With One Line of Code," *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2017-October, pp. 5562–5570, Apr. 2017, doi: 10.1109/ICCV.2017.593.

[42] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLO, GitHub repository," 2023. Accessed: Nov. 11, 2026. [Online]. Available: https://github.com/ultralytics/ultralytics

[43] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush, "Transformers: State-of-the-Art Natural Language Processing," *EMNLP 2020 - Conf. Empir. Methods Nat. Lang. Process. Proc. Syst. Demonstr.*, pp. 38–45, 2020, doi: 10.18653/V1/2020.EMNLP-DEMOS.6.

[44] A. A. Taha and A. Hanbury, "Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool," *BMC Med. Imaging*, vol. 15, no. 1, pp. 29-, Aug. 2015, doi: 10.1186/S12880-015-0068-X.

[45] B. Efron and R. J. Tibshirani, "An Introduction to the Bootstrap," *Chapman and Hall/CRC, May 1994*, doi: 10.1201/9780429246593.

[46] F. Wilcoxon, "Individual Comparisons by Ranking Methods," Biometrics Bulletin, vol. 1, no. 6, pp. 80–83, 1945, doi: 10.2307/3001968.

[47] S. Holm, "A Simple Sequentially Rejective Multiple Test Procedure," *Scand. J. Stat.*, vol. 6, no. 2, pp. 65–70, 1979, doi: 10.2307/4615733.



Copyright © by authors and 50Sea. This work is licensed under Creative Commons Attribution 4.0 International License.