Supplementary Materials for Bones Can't Be Triangles: Accurate and Efficient Vertebrae Keypoint Estimation through Collaborative Error Revision

Summary. This supplementary material enriches the main manuscript by providing comprehensive details of our methodology, additional visualizations, and extended experimental results. Section A presents additional experimental analyses, including a study of user click distribution, the integration of KeyBot with various interaction models, training KeyBot with real keypoint errors, annotation time comparison, and sensitivity analysis. Section B offers an in-depth description of our approach, outlining both its conceptual framework and algorithm. Section C elaborates on the implementation details, experimental setups, and baselines used for comparison. Section D investigates the application of KeyBot in a context where multiple refinement paths are explored, and the best path is chosen by the user, demonstrating its enhanced utility and effectiveness in

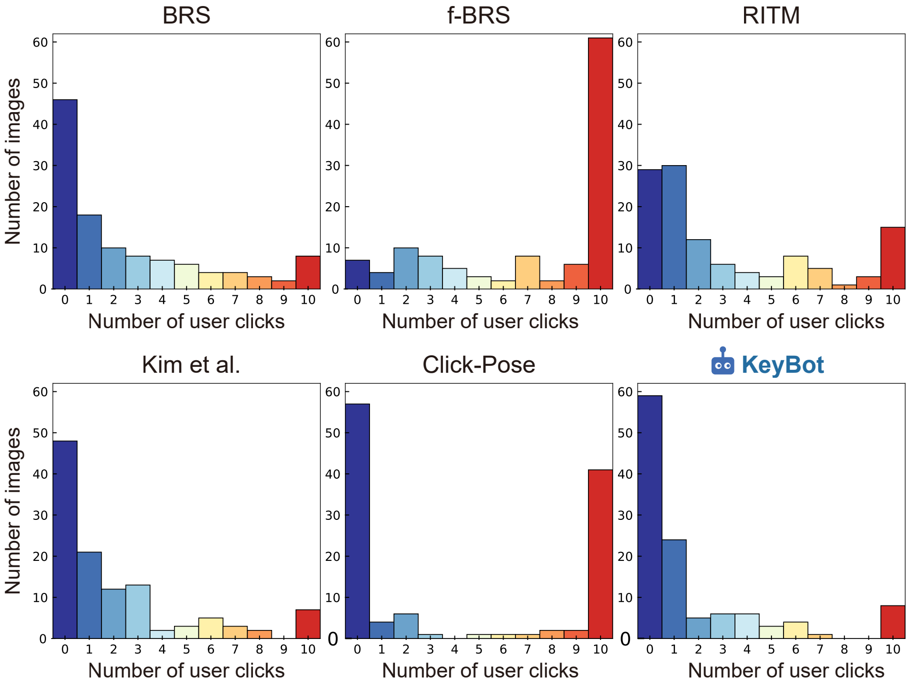


Fig. 9: Distribution of the number of user clicks (NoC) necessary to reach a target mean radial error (MRE) of 20 on the AASCE dataset.

the keypoint annotation process. Section E discusses the limitations and broad impact of our work. Section F provides additional qualitative results, further affirming the practicality and impact of our work.

A More experimental results

Section A.1 analyzes the distribution of the number of user clicks. Section A.2 explores the integration of KeyBot with various interaction models. Section A.3 investigates training KeyBot with real keypoint errors. Section A.5 conducts a sensitivity analysis on the detector.

A.1 Comparative analysis of user clicks required for target accuracy

In this analysis, we examine the distribution of the number of user clicks (NoC) required to achieve a target mean radial error (MRE) of 20, denoted as NoC_{10}@20. We assess the performance of our proposed method, KeyBot, which operates with a maximum of three iterations, in comparison to various baseline models, as shown in Fig. 9. Notably, KeyBot reaches the target MRE with zero user clicks for a significant proportion of images, outperforming the baseline models. In more than half of the entire instances, the images achieve the required accuracy autonomously, demonstrating KeyBot's efficiency in preemptively correcting major errors. KeyBot predominantly achieves the target MRE with fewer user interactions, reinforcing its effectiveness in reducing user effort while maintaining high accuracy in vertebrae keypoint estimation.

A.2 Enhancing interactive keypoint estimation with keyBot

We rigorously assess KeyBot by integrating it with two interactive keypoint estimation frameworks, the model proposed by Kim et al. [10] and Click-Pose [26]. The results, shown in Table 5, include:

Manual revision (`manual`). Manual correction refers to user adjustments on the initial predictions of Kim et al. and Click-Pose without any subsequent model modification, assessing error reduction achieved solely through user revision.

Model revision. The model revision results in gray show improvements due to automated refinements based on user feedback. The difference from manual revisions quantifies the error reduction attributable to the interaction model.

KeyBot. The KeyBot results in blue demonstrate its performance when added to existing models. The same KeyBot instance integrates seamlessly with either framework, requiring no additional training.

KeyBot without accumulating false predictions (`w/o fp`). The results show the performance of KeyBot without the proposed false prediction accumulation strategy, specifically with the Kim et al. model, because this aspect is not applicable to Click-Pose (refer to Section C.4 for more details).

Our findings reveal that adopting KeyBot consistently improves the keypoint estimation performance of baseline models across all three datasets, demonstrating its robustness and effectiveness. KeyBot significantly reduces errors in

Table 5: Performance comparison of mean radial error in keypoint estimation across the AASCE, BUU-AP, and BUU-LA datasets. `UC` denotes the count of user clicks provided to the model.

Method	Interaction backbone	AASCE					BUU-AP			BUU-LA		
		-	`UC1`	`UC2`	`UC3`	`UC4`	-	`UC1`	`UC2`	-	`UC1`	`UC2`
Kim et al. [10]	`manual`	51.58	49.73	48.07	46.49	44.98	42.31	38.65	35.48	23.43	20.81	18.63
Kim et al. [10]	Kim et al.	51.58	30.60	25.78	21.60	19.08	42.31	23.26	16.46	23.43	14.29	10.29
KeyBot-`i3` `w/o fp`	Kim et al.	43.54	27.15	23.36	19.34	16.63	**31.85**	**20.65**	15.87	18.97	**13.34**	9.00
KeyBot-`i1`	Kim et al.	44.18	**25.93**	**20.66**	**18.03**	16.37	32.01	21.79	15.97	18.77	13.39	9.12
KeyBot-`i2`	Kim et al.	42.52	27.03	22.95	18.72	**16.23**	31.88	**20.65**	**15.81**	19.11	13.47	9.03
KeyBot-`i3`	Kim et al.	**41.70**	26.59	25.02	21.23	16.76	31.87	20.66	15.84	**18.74**	13.36	**8.97**
Click-Pose [26]	`manual`	54.65	52.80	51.27	49.85	48.48	32.72	29.25	26.13	33.70	30.02	26.88
Click-Pose [26]	Click-Pose	54.65	46.50	44.08	41.73	40.04	32.72	29.30	26.20	33.70	21.62	17.38
KeyBot-`i1`	Click-Pose	52.62	46.10	43.65	41.17	39.39	31.66	28.03	24.50	33.98	20.53	**16.87**
KeyBot-`i2`	Click-Pose	51.38	45.98	43.53	40.98	39.09	**31.45**	**27.91**	**24.27**	33.29	20.31	16.97
KeyBot-`i3`	Click-Pose	**51.24**	**45.92**	**43.48**	**40.89**	**38.87**	31.56	28.01	24.35	**32.39**	**20.19**	17.14

Table 6: Comparison of real and synthetic errors on the AASCE dataset.

Training		Mean radial error					NoC_{10}	NoC_{10}	NoC_{10}	NoC_{10}
syn	real	-	`UC1`	`UC2`	`UC3`	`UC4`	@20	@30	@40	@50
✗	✗	51.58	30.60	25.78	21.60	19.08	2.10	1.54	1.23	1.03
✓	✗	**41.70**	**26.59**	**25.02**	21.23	**16.76**	**1.74**	**1.32**	**0.94**	**0.68**
✓	✓	46.42	28.64	26.09	**19.89**	17.80	1.88	1.34	0.99	0.72
✗	✓	51.55	30.64	26.44	21.14	18.37	2.11	1.52	1.28	1.01

scenarios with no user interaction. For instance, on the AASCE dataset, KeyBot notably lowers MREs, a trend also observed in the BUU-AP and BUU-LA datasets. The model demonstrates further error reduction post user interactions. Its variant (`w/o fp`) exhibits slightly lower, yet still notable, performance. Overall, the results demonstrate that KeyBot is model-agnostic and can be adapted to different frameworks without the need for model-specific training.

A.3 Training KeyBot with real keypoint errors

We experiment with including real keypoint mistakes in the KeyBot training dataset, using a probability distribution of 40% real mistakes, 40% synthetic errors, and 20% accurate keypoints. However, including real errors resulted in decreased performance, as shown in Table 6. Real errors have high variability and lack consistent patterns, making it challenging to identify a clear pattern to correct. In contrast, synthetic errors are clearly defined and consistent, facilitating better convergence during training and leading to improved performance.

Table 7: Comparison of annotation cost per image on the BUU-LA dataset.

Method	Time (s)	User click
manual revision	23.10 ± 2.93	7.02 ± 1.74
model revision w/o KeyBot	21.61 ± 12.66	3.04 ± 1.94
model revision w/ KeyBot	**11.71** ± 2.23	**0.44** ± 0.30

Table 8: Sensitivity analysis of the detector. MRE is measured on the BUU-AP and BUU-LA datasets. k and s denotes represents the number of keypoints examined simultaneously and the stride during inference, respectively. UC denotes the number of user clicks. KeyBot-i3 is used for the analysis.

Detector		BUU-AP			BUU-LA		
k	s	-	UC1	UC2	-	UC1	UC2
68	-	39.32	22.39	19.64	20.44	13.94	9.53
8	1	31.87	18.30	16.02	**18.61**	14.87	**8.94**
8	2	34.85	18.70	15.97	18.64	13.44	8.96
8	3	**31.75**	**18.25**	15.94	18.74	13.44	8.99
8	4	31.87	20.66	**15.84**	18.74	**13.36**	8.97
4	1	36.17	20.25	16.24	18.79	13.42	8.99
4	2	32.22	20.86	16.34	**18.42**	**13.39**	9.07
4	3	**31.74**	**18.14**	**16.03**	18.56	13.46	10.07
4	4	32.27	22.94	16.14	18.61	15.10	**8.95**

A.4 Comparison of annotation time

We conduct a user study with 15 participants, each tasked with annotating 22 keypoints on ten challenging radiographs from the BUU-LA dataset [11]. Participants are divided into three groups: one using KeyBot, one without it, and one using only initial model predictions without subsequent model revision. As summarized in Table 7, the results show that KeyBot significantly reduces annotation time and user clicks, demonstrating its efficiency. Although the computation time for KeyBot is higher, its average inference time remains under 0.22 seconds, which is negligible.

A.5 Sensitivity analysis of the detector in keypoint error detection

We conduct a sensitivity analysis on the detector, as detailed in Table 8. We investigate the impact of varying the number of simultaneously examined keypoints (k) and the stride (s) during inference on keypoint estimation accuracy. We observe a substantial decrease in performance when the detector assesses all keypoints at once, resulting in the most significant keypoint estimation errors. This performance decline suggests that analyzing the entire bone structure

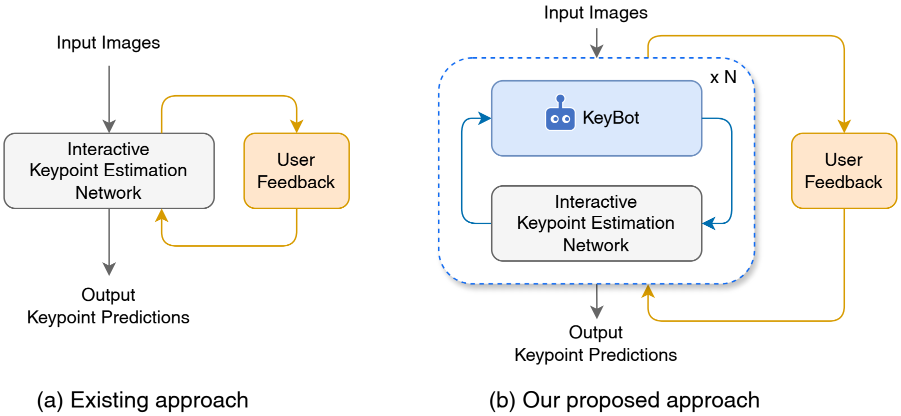

Input Images
Interactive Keypoint Estimation Network
User Feedback
Output Keypoint Predictions
(a) Existing approach
Input Images
x N
KeyBot
Interactive Keypoint Estimation Network
User Feedback
Output Keypoint Predictions
(b) Our proposed approach

Fig. 10: Comparison between the (a) existing interactive keypoint estimation framework and (b) the proposed approach adopting KeyBot.

at once increases complexity, diminishing the detector's effectiveness. Focusing on specific bone segments during detection proves to be more efficient. KeyBot maintains robust performance with different input keypoint numbers, especially when $k = 4$ and $k = 8$.

Additionally, we analyze the effect of varying the stride s with a fixed keypoint window k. KeyBot performs consistently across different stride settings. Particularly with an input keypoint number of four, smaller stride values than k slightly enhance performance. A smaller stride provides a more detailed and contextually varied examination of each keypoint, improving the detector's accuracy and reliability.

B Additional details of KeyBot

This section complements an explanation of KeyBot. First, Section B.1 describes the detailed conceptual framework of our method, and Section B.2 offers an algorithm that elucidates the overall process of KeyBot.

B.1 Overview of the proposed approach

Existing interactive keypoint estimation approaches [10,26] operate by predicting keypoints from images and refining these predictions based on user feedback, as depicted in Fig. 10(a). However, they lack the capability for self-correction without user intervention.

To address this limitation, we introduce KeyBot, which independently evaluates the interaction model's predictions, pinpointing and correcting errors autonomously, as illustrated in Fig. 10(b). This approach facilitates autonomous correction of predictions, thereby enabling users to concentrate on subsequent refinements after KeyBot's preliminary error rectification.

Why a user-interactive approach? In medical imaging, the accuracy of keypoint estimation models is paramount. Despite recent advancements, human adjustments are often needed to address inherent model biases and errors. However, manual revision, while necessary to ensure reliability and accuracy, is often a tedious and time-consuming task. Thus, an interactive approach incorporating human feedback aims to streamline this process by refining inaccuracies with minimal user intervention.

Why KeyBot? Existing interactive models require users to initiate error assessment, which can be laborious and error-prone, especially with numerous predictions or clustered inaccuracies. KeyBot autonomously conducts an initial assessment, addressing basic errors such as fundamental misidentifications of bone structures. By addressing these primary errors, it reduces the time users spend on basic error corrections, allowing for a more focused and efficient use of human expertise.

Why not end-to-end? While an interactive keypoint estimation model learns to correct its errors based on user feedback, it lacks explicit training for specific error types. In contrast, KeyBot is designed to identify and correct three specific error types, utilizing error simulation in its training phase. Its independent structure prevents it from inheriting potential biases or limitations of the interaction model, ensuring an objective and reliable keypoint estimation process.

Ongoing collaborative loop Integrating KeyBot within the interactive keypoint estimation framework in a feedback-providing manner maintains an iterative and collaborative loop. This loop ensures that the refinement process is not a one-off correction but a continuous improvement cycle, accommodating further refinements and adjustments from both KeyBot and human users, leading to more accurate outcomes in medical image analysis.

B.2 Algorithm of KeyBot

The complete procedure of our proposed approach is encapsulated in Algorithm 1. Given an input image, the interaction model makes an initial keypoint prediction, followed by two sequential phases: the KeyBot phase (with N iterations) and the user phase (one iteration). During the KeyBot phase, KeyBot performs a preliminary step to correct errors and generates corrective feedback. This feedback is fed into the interaction model, akin to user feedback. The interaction model then generates a new, corrected prediction. This phase repeats for N iterations or until KeyBot detects no errors. Subsequently, in the user phase, the user corrects a single error, and the interaction model updates its results accordingly. This entire process repeats until it reaches the maximum number of user interactions, T.

Algorithm 1: Inference with KeyBot

```
Data: input image $\boldsymbol{x}$, maximum user click $T$, maximum KeyBot
      iteration number $N$, interaction model $\mathcal{F}_\theta$, KeyBot detector $\mathcal{G}_{\phi_r}$,
      KeyBot corrector $\mathcal{G}_{\phi_s}$
Result: $\boldsymbol{y}$, the final keypoint prediction
$\boldsymbol{c}_{0,0} = 0, \boldsymbol{e}_{0,0} = 0$
$\boldsymbol{y}_{0,0} \leftarrow \mathcal{F}_\theta(\boldsymbol{x}, \boldsymbol{c}_{0,0}, \boldsymbol{e}_{0,0})$  ▷ initial interaction model forward
while $t < T$ do
    while $n < N$ do
        $\boldsymbol{\nu}_{t,n+1} \leftarrow \mathcal{G}_{\phi_r}(\boldsymbol{x}, \boldsymbol{y}_{t,n})$  ▷ KeyBot detector
        if $|\boldsymbol{\nu}_{t,n+1} \setminus \boldsymbol{\rho}_t| = 0$ then
            break
        else
            $\boldsymbol{z}_{t,n+1} \leftarrow \mathcal{G}_{\phi_s}(\boldsymbol{x}, \boldsymbol{y}_{t,n})$  ▷ KeyBot corrector
            $\boldsymbol{c}^i_{t,n+1} \leftarrow \boldsymbol{z}^i_{t,n+1}, i \in \boldsymbol{\nu}_{t,n+1} \setminus \boldsymbol{\rho}_t$
            $\boldsymbol{e}^i_{t,n+1} \leftarrow \boldsymbol{y}^i_{t,n}, i \in \boldsymbol{\nu}_{t,n+1} \setminus \boldsymbol{\nu}_{t,:n} \setminus \boldsymbol{\rho}_t$
            $\boldsymbol{y}_{t,n+1} \leftarrow \mathcal{F}_\theta(\boldsymbol{x}, \boldsymbol{c}_{t,n+1}, \boldsymbol{e}_{t,n+1})$  ▷ interaction model forward
        end
    end
    $\boldsymbol{c}_{t+1,0} \leftarrow 0, \boldsymbol{e}_{t+1,0} \leftarrow \boldsymbol{e}_{t,n+1}$
    $\boldsymbol{\rho}_{t+1}, \boldsymbol{u}_{t+1} \leftarrow \Psi(\boldsymbol{x}, \boldsymbol{y}_{t,n+1}, \boldsymbol{u}_t)$  ▷ User revision
    $\boldsymbol{c}^i_{t+1,0} \leftarrow \boldsymbol{u}^i_{t+1}, i \in \boldsymbol{\rho}_{t+1}$
    $\boldsymbol{e}^i_{t+1,0} \leftarrow \boldsymbol{y}^i_{t,n+1}, i \in \boldsymbol{\rho}_{t+1}$
    $\boldsymbol{y}_{t+1,0} \leftarrow \mathcal{F}_\theta(\boldsymbol{x}, \boldsymbol{c}_{t+1,0}, \boldsymbol{e}_{t+1,0})$  ▷ interaction model forward
end
```

C Experimental details

This section provides comprehensive details about the implementation details (Section C.1), dataset descriptions (Section C.2), metric definitions (Section C.3), and reproducibility details for baseline models (Section C.4).

C.1 Implementation details of KeyBot

We describe a detailed experimental setup of our approach: the detector, the corrector, and the interaction model. Also, we elaborate on the error simulation methods employed in training the detector and the corrector, including vertex misidentification (misvertex), bone misidentification (misbone), and left-right inversion (lr-inversion).

Detector The detector analyzes eight ($k = 8$) keypoints simultaneously, classifying each as accurate or inaccurate. Input X-ray images, cropped and resized around these keypoints to 128 × 128 dimensions, are concatenated with Gaussian keypoint heamtaps. The detector evaluates the abnormality likelihood for each keypoint using a sigmoid function. Training labels are generated by marking synthetically displaced keypoints as one (indicative of errors) and the rest as zero. During inference, the detector iteratively processes keypoints with a stride of four ($s = 4$). On the BUU-LA datasets, this process is applied to only 20 keypoints, excluding the final two for error detection. Any keypoint with an anomaly probability above 0.5 is flagged as erroneous.

The detector employs a modified ResNet-18 [7] architecture, adapted for combined image and heatmap inputs. The training process incorporates simulated vertex misidentification errors, displacing up to three keypoints per image for AASCE and four keypoint for BUU-AP and BUU-LA datasets, respectively. Selected keypoints shift up to four indices away from their original position, with wrapping for out-of-range indices. The training utilizes Binary Cross-Entropy (BCE) loss over 300 epochs with early stopping (zero patience) and an AdamW optimizer with a learning rate of 0.001.

Corrector The corrector processes the entire image, resized to 256×128, alongside Gaussian keypoint heatmaps of matching resolution. It generates reconstructed keypoint locations as heatmaps, using a sigmoid function in the final layer. The architecture is based on DeepLab-v3 with a ResNet50 encoder. During training, KeyBot is trained on accurate keypoints with a 20% probability and simulated errors with an 80% probability. Training uses three error types or accurate keypoints, with varied probabilities for each dataset (Please refer to the source code for more details).

(1) **Vertex misidentification errors**: Up to nine keypoints are displaced, with a multinomial probability distribution for the number of keypoints to shift. Keypoints are selected randomly with equal probability, and they are shifted to maximally four indices away from their original index.
(2) **Bone misidentification errors**: To simulate misbone errors, movement type (up, down, or accurate) is selected with equal probability. For the shifts, there is an equal probability for each of the following scenarios: moving all keypoints from the first to the last, moving keypoints from a random starting point to the last keypoint, moving keypoints from the first to a random end point, and moving keypoints between a random start and end point. For keypoints located at either the first or last vertebrae, relocation may occur outside the targeted bone structure, with the magnitude determined by positional differences between either the first and second vertebrae or the last and its immediate predecessor.
(3) **Left-right inversion errors**: Every left-right pair is independently swapped with a 90% probability.

Similar to the detector, the corrector is trained over 300 epochs with early stopping, using an AdamW optimizer with the learning rate of 0.001.

Table 9: Summary of X-ray image datasets for keypoint estimation used in our work.

Dataset	Target keypoints	Number of images				Human annotation error			
		Total	Train	Val	Test	Total	misvertex	misbone	lr-inversion
AASCE [24]	68	563	325	122	116	45	18	27	-
BUU-AP [11]	20	399	240	80	79	1	-	-	1
BUU-LA [11]	22	397	237	80	80	3	-	-	3

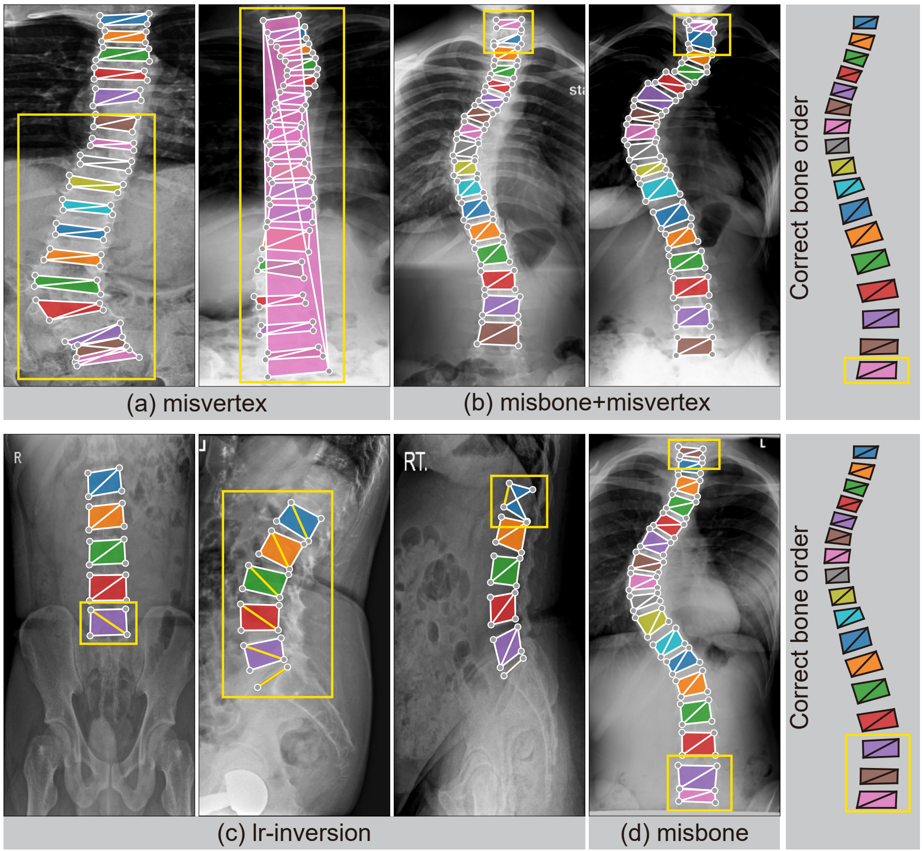


Fig. 11: Errors in human annotation present in the AASCE dataset in (a),(b) and (d), and in the BUU-AP and BUU-LA datasets in (c). A diagonal line connecting the upper-right vertex to the lower-left vertex is indicated in each vertebra.

Interaction model The interaction model estimates keypoint heatmaps from input X-ray images with a size of 512 × 256. We extend the model proposed by Kim et al. [10] by integrating revision feedback from either KeyBot or user input and adopting accumulated false predictions. Training includes iterative refinement based on simulated user feedback, providing groundtruth keypoint locations as user revision. During inference, the most significant keypoint error is modified to simulate user interaction.

C.2 Datasets

In our study, we utilize public X-ray image datasets, namely AASCE, BUU-AP, and BUU-LA, as detailed in Table 9. An extensive analysis of these datasets reveals three primary types of human annotation errors: vertex misidentification (misvertex), bone misidentification (misbone), and left-right inversion (lr-inversion). The AASCE dataset predominantly exhibits misvertex and misbone errors, whereas the BUU-AP and BUU-LA datasets mainly exhibit lr-inversion errors. To ensure the integrity and reliability of our evaluation, images with such critical annotation errors were excluded, as illustrated in Fig. 11.

AASCE [24] The AASCE dataset consists of 608 anterior-posterior X-ray images, annotated with 68 keypoints across 17 vertebrae. We follow the original train-test split, further partitioning the training set for validation purposes. Error analysis identifies that it contains nine misvertex and 18 misbone errors in the training set, one misvertex and five misbone errors in the validation set, and eight misvertex and four misbone errors in the test set, as exemplified in Fig. 11(a,b,d). Two cases showing a mixture of misvertex and misbone errors are counted as misbone errors.

BUU-AP [11] Comprising 400 anterior-posterior view X-ray images, BUU-AP dataset is annotated with 20 keypoints per image. The image sizes range from 1434 × 1072 to 3072 × 3040. We randomly split the images into training, validation, and test sets, identifying a lr-inversion error in the test set, as shown in Fig. 11(c).

BUU-LA [11] The BUU-LA dataset includes 400 left lateral view X-ray images, each with 22 annotated spinal keypoints with image sizes ranging from 1956×968 to 3072 × 3040. It undergoes similar partitioning as BUU-AP, with three lr-inversion errors identified in the training set, as exemplified in Fig. 11(c).

C.3 Definition of mean radial error

In our study, we assess the keypoint estimation accuracy using the mean radial error (MRE). MRE measures the average Euclidean distance between the predicted and groundtruth coordinates of keypoints. Specifically, for each of the K target keypoints in a sample, let $\boldsymbol{p}_i^*$ denote the groundtruth coordinates and $\boldsymbol{p}_i$ the predicted coordinates of the i-th keypoint. The MRE is calculated as:

$$\mathrm{MRE} = \frac{1}{K} \sum_{i=1}^{K} ||\boldsymbol{p}_i^* - \boldsymbol{p}_i||_2. \tag{1}$$

MRE measures the overall precision of keypoint estimation results.

C.4 Reproducibility for baselines

Interactive segmentation models. In this work, we compare our method with several interactive segmentation models, including BRS [8][1], f-BRS [21][2], and RITM [22][3], for our keypoint estimation task. Using their official source codes, we modify these models to produce outputs for the specific number of keypoints required in our study, aligning with the hyperparameter settings from Kim et al. [10] for consistency.

Kim et al. [10] We adhere to the experimental settings outlined in the official source code[4]. For experiments on the AASCE dataset, we utilize their pretrained network. In the case of the BUU-AP and BUU-LA datasets, we train the model using identical hyperparameters as those used for the AASCE dataset, with the only modification being the selection of keypoint subsets. This adjustment is necessary to accommodate the morphology-aware loss proposed in the study. When integrating KeyBot, we handle KeyBot's revision feedback as user interaction input, similar to processing user clicks.

Click-Pose [26] We utilize the official source code of Click-Pose[5], maintaining the hyperparameters consistent across all datasets, while adjusting the number of keypoints as required. During the training phase, we exclude the Object Keypoint Similarity (OKS) loss which is specifically tailored for human pose estimation. As a result, Click-Pose training involves using a combination of L1 loss for bounding boxes, intersection over union (IOU) loss, classification loss, and L1 loss for keypoints. Click-Pose employs a keypoint regression-based approach, directly estimating the coordinates of keypoints. For integration with KeyBot, we feed KeyBot's revision feedback into the human-to-keypoint decoder module of Click-Pose, akin to handling user clicks. Given that Click-Pose does not employ a heatmap-based keypoint estimation approach, we do not incorporate false predictions into the input framework.

D Exploring multiple refinement paths of KeyBot

This section introduces a novel collaborative annotation approach involving KeyBot, the user, and the interaction model. We investigate the application of KeyBot in a context where multiple refinement paths are explored, and the best path is chosen by the user, demonstrating its enhanced utility and effectiveness in the keypoint annotation process. This method involves multiple iterative interactions between KeyBot and the backbone model, yielding a variety of refined results. These results, alongside the initial prediction, are presented to the

[1] https://github.com/wdjang/BRS-Interactive_segmentation
[2] https://github.com/saic-vul/fbrs_interactive_segmentation/tree/master
[3] https://github.com/SamsungLabs/ritm_interactive_segmentation
[4] https://github.com/seharanul17/interactive_keypoint_estimation
[5] https://github.com/IDEA-Research/Click-Pose

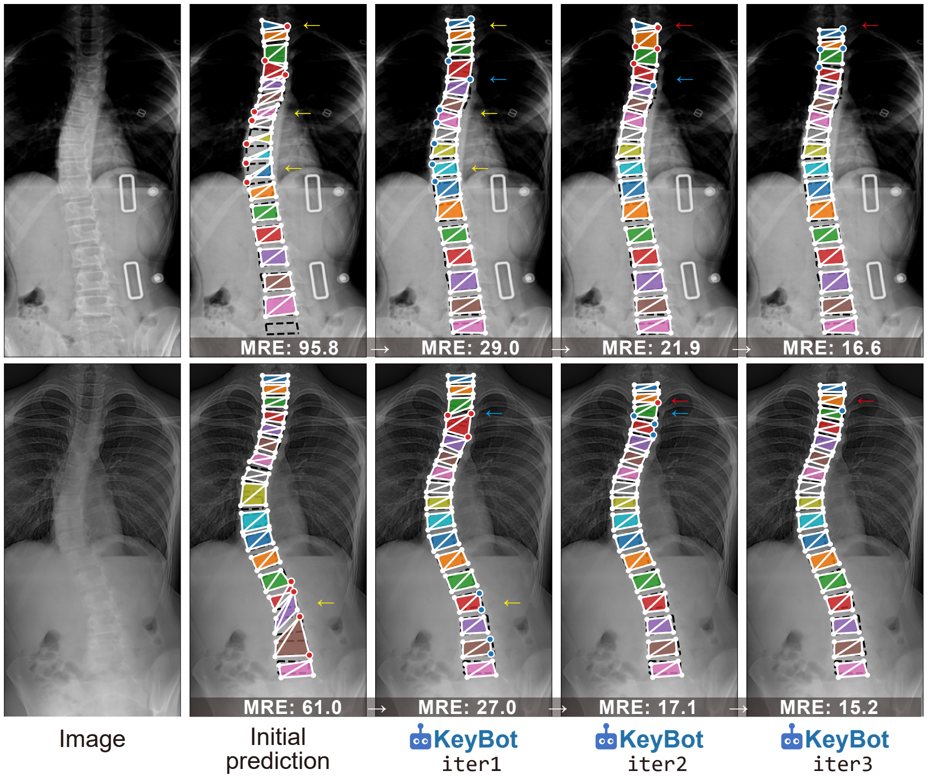


Fig. 12: Iterative refinement results of KeyBot on the AASCE dataset. In the first example, KeyBot corrects the upper and middle part in the image indicated by the yellow arrow, lowering the bones overall in the first iteration. It revises the blue arrow area in the second iteration and finally lowers the incorrectly positioned bones at the top in the third iteration. In the second example, the first iteration adjusts the lower part as indicated by the yellow arrow. In the second iteration, the front end of the bones marked in blue is lowered overall. Finally, in the third iteration, the model properly adjusts the position of the bones at the top, marked in red. A diagonal line connecting the upper-right vertex to the lower-left vertex is indicated in each vertebra.

user for selection and potential further refinement. Among multiple refinement iterations, users can select the most precise result as their foundation for any additional modifications. Users also have the option to ignore the model updates and correct them manually if needed.

This approach resembles the concept of beam search, where each KeyBot refinement iteration represents a node in the search space. We allow users to explore multiple paths, i.e., KeyBot iterations, and to choose the best path. Providing multiple iteration results is feasible because the average inference time for the interaction model is 0.181 seconds, and for KeyBot is 0.216 seconds, which is negligible. The procedure consists of three steps. Initially, the backbone model

Table 10: Performance comparison of mean radial error in keypoint estimation across the AASCE, BUU-AP, and BUU-LA datasets. For KeyBot-`i3 w/ beam search`, the best prediction is chosen among three iterations. No user input is provided.

Method	AASCE	BUU-AP	BUU-LA
BRS [8]	45.65	51.22	40.20
f-BRS [21]	64.06	44.05	36.03
RITM [22]	56.03	36.43	23.27
Kim et al. [10]	51.58	42.31	23.43
Click-Pose [26]	54.65	32.72	33.70
KeyBot-`i1`	44.18	32.01	18.77
KeyBot-`i2`	42.52	31.88	19.11
KeyBot-`i3`	41.70	31.87	18.74
KeyBot-`i3 w/ beam search`	**39.29**	**31.43**	**18.70**

produces an initial keypoint prediction from the input X-ray image, forming the baseline prediction. Next, KeyBot evaluates this prediction, identifies inaccuracies, and makes corrections, leading to the first refined set of keypoints. This iteration repeats, resulting in multiple refined sets. Lastly, the user reviews all keypoint sets, including the initial and refined ones, and selects the most accurate set as a basis for any further adjustments.

The primary benefit of this approach is that it provides users with multiple refined results in addition to the initial prediction, as shown in Fig. 12. This allows for a well-informed decision-making process, where users select the most precise result as their foundation for any additional modifications.

Additionally, we analyze the error reduction achieved through this multi-iteration presentation approach, as shown in Table 10. This multi-iteration presentation approach with KeyBot, combining KeyBot's iterations with user selection and refinement, optimizes prediction accuracy while significantly reducing user effort.

E Discussion

The anatomical complexity and similarity among vertebrae make vertebrae keypoint estimation prone to errors that require substantial human efforts to correct. By advancing methodology to enhance accuracy and efficiency in this area, our work represents a substantial technical contribution. However, a limitation is that its correction precision occasionally falls short of human-level precision when most keypoints contain substantial errors, as shown in Fig. 8 of our main manuscript. In these cases, KeyBot disregards the erroneous predictions and generates completely new ones. However, because KeyBot generates revisions based on the initial predictions (refer to Fig. 4 of our main manuscript), it is influenced by these errors, limiting its ability to provide precise feedback for incorrect vertebrae shapes. This necessitates user feedback to guide further precise adjustments. Future work aims to develop a more robust feedback mechanism to address these cases effectively, enhance accuracy, and minimize user input.

Although our approach is developed specifically for vertebrae, it introduces a general framework for addressing domain-specific errors that can be extended to other fields. By characterizing domain-specific error types and generating synthetic data to represent these inaccuracies, our approach can be used to develop an auxiliary model to detect and correct them efficiently. This adaptability highlights the broader impact and versatility of our work.

F Additional qualitative results

This section presents additional qualitative results on the AASCE dataset, as depicted in Figs. 13 and 14, and on the BUU-AP and BUU-LA datsets, as shown in Fig. 15, highlighting the effectiveness of KeyBot in autonomously revising multiple keypoints simultaneously, minimizing the need for user intervention. A diagonal line connecting the upper-right vertex to the lower-left vertex is indicated in each vertebra.

Initial model predictions often exhibit significant inaccuracies in representing bone shape, posing challenges for users in identifying and correcting errors, mainly when keypoints are densely clustered or overlap. KeyBot significantly streamlines this error correction process by effectively identifying erroneous keypoints, which are marked with red dots in the initial predictions. It distinguishes well-represented bone shape keypoints from severe morphological distortions, including misvertex errors (incorrect positioning of a portion of vertebra keypoints) and lr-inversion errors (incorrect left-right orientation). KeyBot is also highly effective in recognizing misbone errors, where an entire bone is misidentified, as illustrated in Figs. 13 and 14.

These capabilities allow KeyBot to provide targeted interventions for specific errors, greatly enhancing the accuracy and reliability of the overall keypoint estimation process. By automating the identification and correction of such errors, KeyBot not only improves the precision of the model but also reduces the burden on users, minimizing the cognitive load on users, allowing them to concentrate on verifying and fine-tuning the results, requiring considerably less effort.

For a more comprehensive understanding of KeyBot's capabilities, please refer to our demo video, which visually represents KeyBot's efficiency and accuracy across various examples.

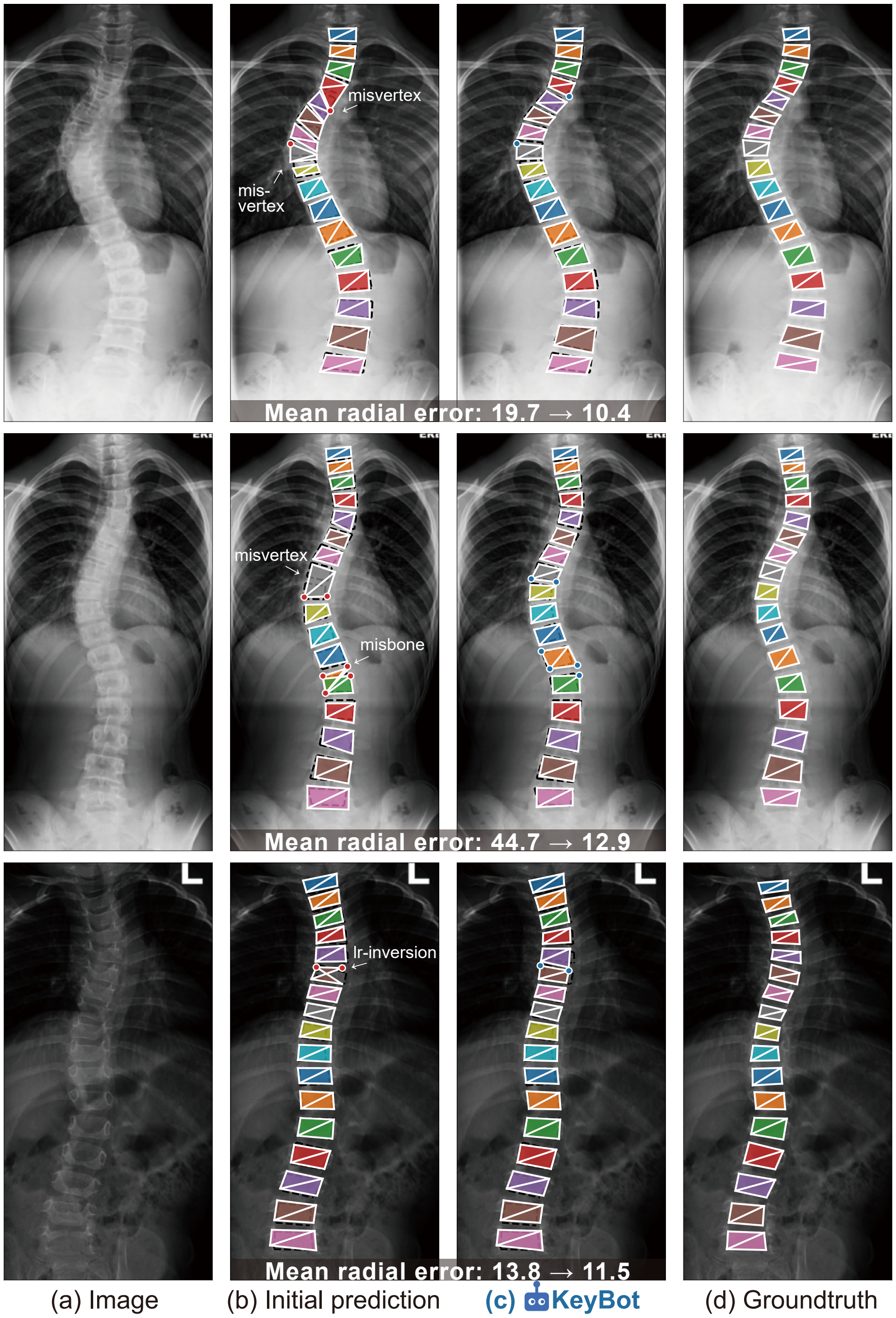


Fig. 13: Additional qualitative results of KeyBot on the AASCE dataset, with a maximum of three iterations.

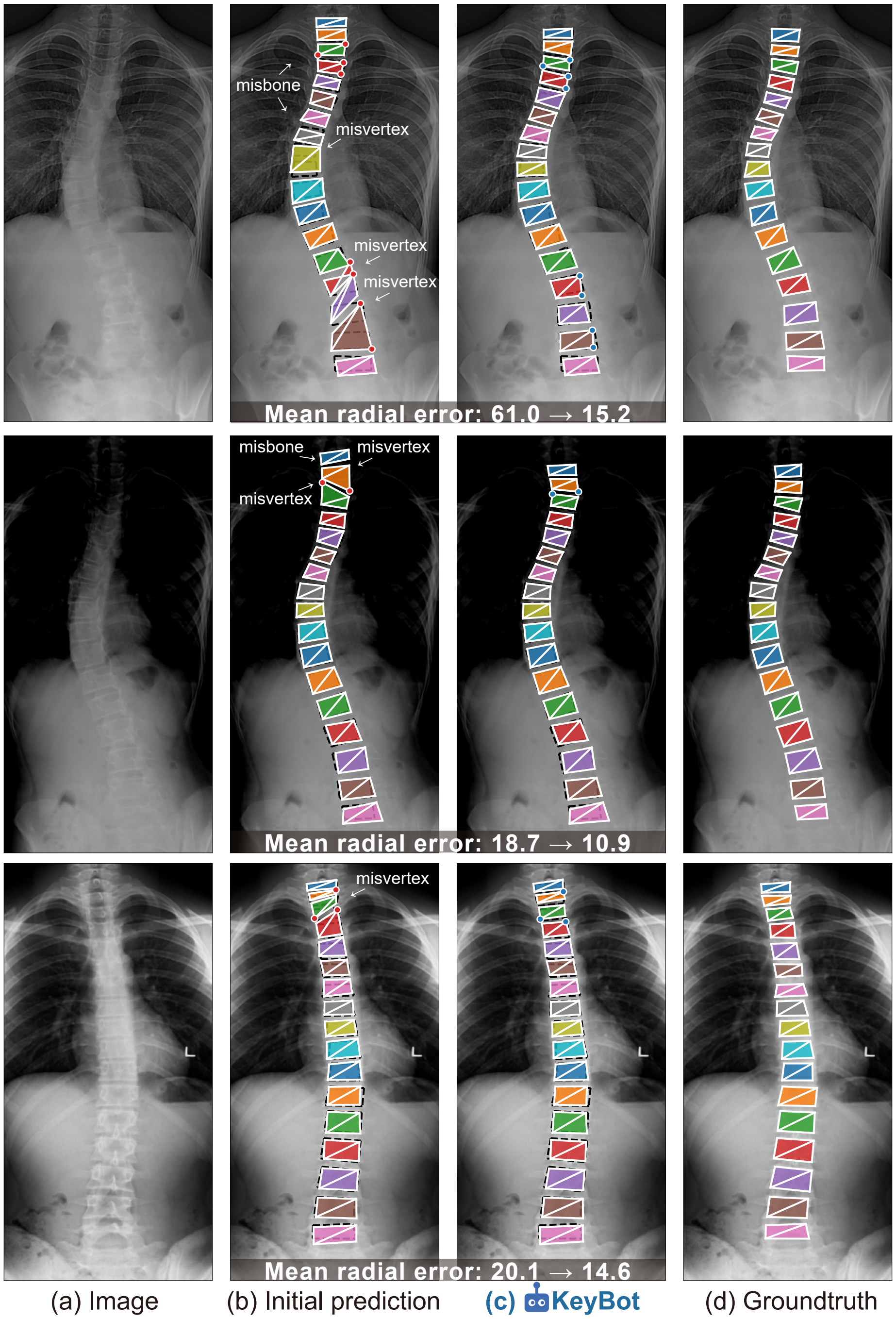
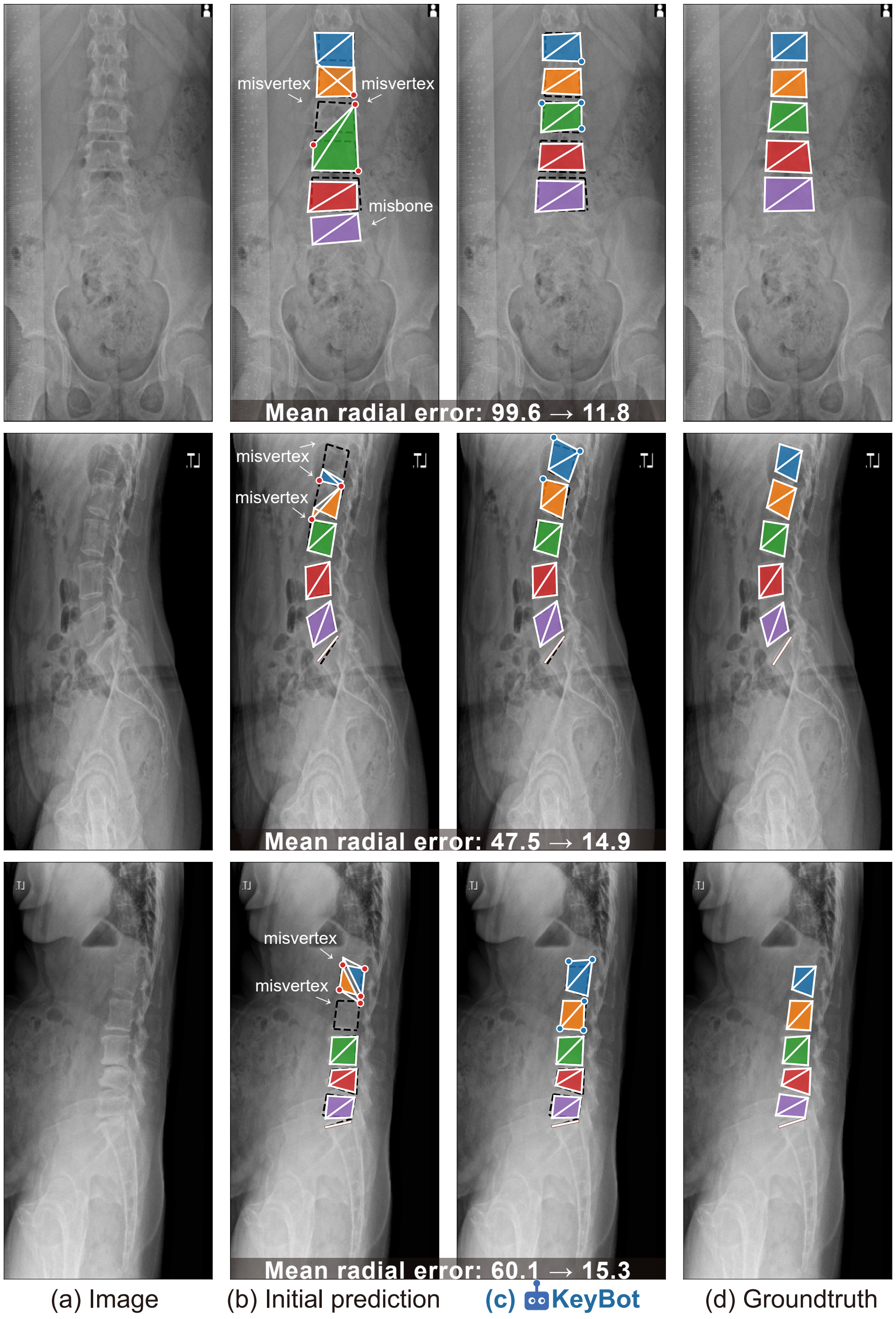


Fig. 14: Additional qualitative results of KeyBot on the AASCE dataset, with a maximum of three iterations.


misvertex
misvertex
misbone
Mean radial error: 99.6 → 11.8
misvertex
misvertex
Mean radial error: 47.5 → 14.9
misvertex
misvertex
Mean radial error: 60.1 → 15.3

(a) Image (b) Initial prediction (c) KeyBot (d) Groundtruth

Fig. 15: Additional qualitative results of KeyBot on the BUU-AP (top) and BUU-LA (middle and bottom) datasets, with a maximum of three iterations.