Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

Challenge Report

Evaluation and comparison of accurate automated spinal curvature estimation algorithms with spinal anterior-posterior X-Ray images: The AASCE2019 challenge

Liansheng Wang[a], Cong Xie[a], Yi Lin[b], Hong-Yu Zhou[c], Kailin Chen[d], Dalong Cheng[d], Florian Dubost[e], Benjamin Collery[e,f], Bidur Khanal[g], Bishesh Khanal[g], Rong Tao[h], Shangliang Xu[h], Upasana Upadhyay Bharadwaj[i], Zhusi Zhong[j], Jie Li[j], Shuxin Wang[a], Shuo Li[k,l,*]

[a] *Department of Computer Science, Xiamen University, Xiamen 361005, China*
[b] *Department of Electronic Information and Communications, Huazhong University of Science, China*
[c] *Tencent YouTu Lab, China*
[d] *iFLYTEK Research South China, Computer Vision Group, China*
[e] *Biomedical Imaging Group Rotterdam, Department of Radiology and Nuclear Medicine, Erasmus MC - University Medical Center Rotterdam, the Netherlands*
[f] *Ecole des Mines de Saint-Etienne, Saint-Etienne, France*
[g] *Institute: NepAl Applied Mathematics and Informatics Institute for Research (NAAMII), China*
[h] *PingAn Technology(shenzhen) Co., Ltd., Shanghai 20000, China*
[i] *University of California, San Francisco*
[j] *School of Electronic Engineering, Xidian University, Xi'an 710071, China*
[k] *Department of Medical Imaging, Western University, ON, Canada*
[l] *Digital Image Group, London, ON, Canada*

ARTICLE INFO

Article history:
Received 4 February 2020
Revised 19 May 2021
Accepted 21 May 2021
Available online 1 June 2021

Keywords:
Scoliosis
Spinal curvature
Challenge
Automated estimation

ABSTRACT

Scoliosis is a common medical condition, which occurs most often during the growth spurt just before puberty. Untreated Scoliosis may cause long-term sequelae. Therefore, accurate automated quantitative estimation of spinal curvature is an important task for the clinical evaluation and treatment planning of Scoliosis. A couple of attempts have been made for automated Cobb angle estimation on single-view x-rays. It is very challenging to achieve a highly accurate automated estimation of Cobb angles because it is difficult to utilize x-rays efficiently. With the idea of developing methods for accurate automated spinal curvature estimation, AASCE2019 challenge provides spinal anterior-posterior x-ray images with manual labels for training and testing the participating methods. We review eight top-ranked methods from 12 teams. Experimental results show that overall the best performing method achieved a symmetric mean absolute percentage (SMAPE) of 21.71%. Limitations and possible future directions are also described in the paper. We hope the dataset in AASCE2019 and this paper could provide insights into quantitative measurement of the spine.

1. Introduction

The spine is the central axis of the human body, with important functions like weight-bearing, shock absorption, protection and movement. It is normally made up of 33 bones called vertebrae, which are subdivided into five regions from top to bottom: cervical (7 vertebrae), thoracic (12 vertebrae), lumbar (5 vertebrae), sacrum (5 vertebrae), and coccyx (4 vertebrae). The upper twenty-four are articulating and separated from each other by intervertebral discs, and the lower nine are fused in adults (Kenneth, 2017).

Scoliosis is a spine condition, characterized by a sideways curve of the spine, accompanied by a rotation of the vertebrae. A normal spine should be straight when viewed from behind and centered on the pelvis. When suffering from scoliosis, the curve is usually "S"- or "C"-shaped over three dimensions (shown in Fig. 1) (Illés et al., 2019). Between 1% and 4% of adolescents have scoliosis,

[*] Corresponding author at: Department of Medical Imaging, Western University, ON, Canada.
E-mail address: slishuo@gmail.com (S. Li).

https://doi.org/10.1016/j.media.2021.102115

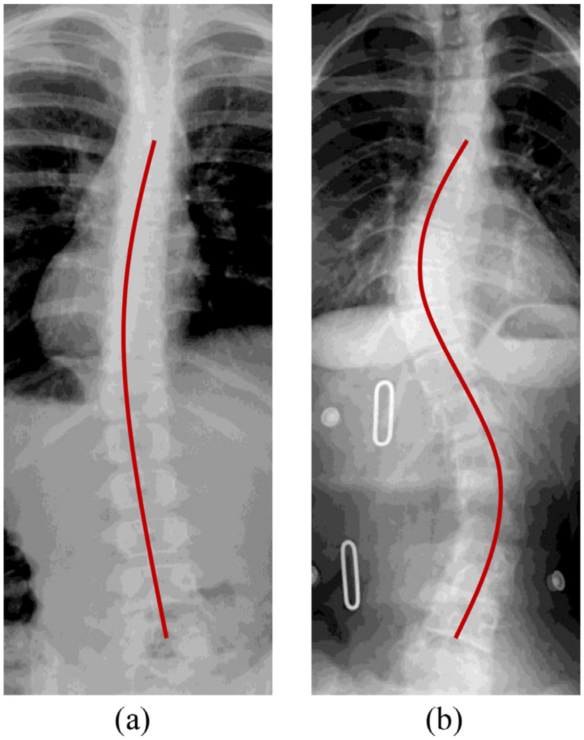

Fig. 1. Examples of scoliosis. (a) “C”-shaped curve, (b) “S”-shaped curve.

Table 1
Scoliosis cases with Cobb angles.

Severity	Cobb angle
Not scoliosis	<10° (Lau, 2013)
Mild scoliosis	10 – 30° (Bloch et al., 2012)
Moderate scoliosis	30 – 45° (Bloch et al., 2012)
Severe scoliosis	>45° (Bloch et al., 2012)

and appears most often during the growth spurt just before puberty (Cheng et al., 2015). Untreated scoliosis may cause long-term sequelae, such as curve progression, back pain, cardiopulmonary issues, and psychosocial concerns (Barton and Weinstein, 2018). Clinically, the diagnosis of scoliosis can be confirmed by x-ray and a subsequent Cobb angle analysis of the images. The Cobb angle is the most common quantification of scoliosis, proposed originally by the American orthopedic surgeon John Robert Cobb (Cobb, 1948). Cobb angle can describe the severity of scoliosis, as shown in Table 1. A Cobb angle between 10 and 30 degrees denotes mild scoliosis. Scoliosis cases with Cobb angles in the range of 30 to 45 degrees are moderate scoliosis. A Cobb angle greater than 45 degrees denotes severe scoliosis. Moreover, people with a Cobb angle greater than 60° usually have respiratory complications (Bloch et al., 2012).

The current gold standard of scoliosis evaluation is the manual Cobb angle measurement endorsed by the Scoliosis Research Society. Clinicians identify the upper and lower endplates of the most tilted vertebrae, then measuring the Cobb angles between them, as shown in Fig. 2. Manual measurement is time-consuming and unreliable. Variations occur between people who do the measuring, as well as between tools used in the process (specifically, the protractor). Thus, accurate automated quantitative estimation of spinal curvature is an essential task for the clinical evaluation and treatment planning of scoliosis.

There have been many challenges in computational methods and clinical applications for spine imaging during the past years (e.g., xVertSeg, IVDM3Seg, and computational challenges on CSI).

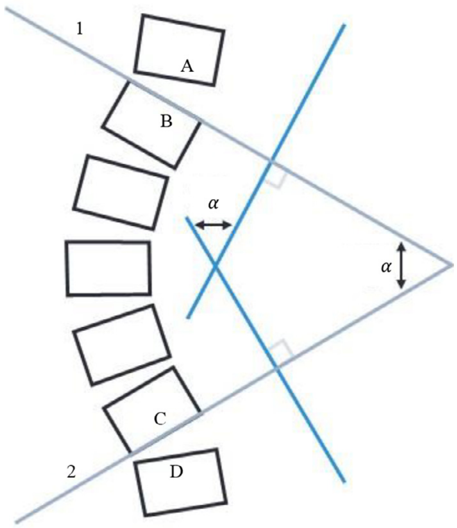

Fig. 2. Cobb angle measurement. α represents the Cobb Angle. First, determine the curved segment and find the top and bottom vertebrae of this curve. The top and bottom vertebrae refer to the vertebrae with the largest inclination toward the concave side of the curved segment. The convex side has a wider intervertebral space, and the first vertebra that begins to widen on the concave side is considered not part of the curve segment (i.e., A, D), so its adjacent first vertebra is considered to be the curved end vertebra (i.e., B, C). Draw a horizontal line on the top edge of the top vertebra (line1), and also draw a horizontal line on the bottom edge of the bottom vertebra (line2). Make a vertical line for each of the two horizontal lines. The angle of the intersection of two vertical lines is the Cobb angle α.

These challenges have allowed objective evaluation and comparison for the methods proposed by participants around the world. For example, the MICCAI-CSI2014 hosted two challenges on “Spine and Vertebrae Segmentation“ and “Vertebrae Localization and Identification“. These challenges mainly focus on the identification of vertebrae or the analysis of vertebral fractures. To date, only a few studies focused on accurate automated spinal curvature estimation (Wang et al., 2019; Zhang et al., 2019; Sun et al., 2017; Aubert et al., 2017). In the AASCE-2019 challenge, in conjunction with MICCAI2019, the objective of the challenge was that researchers were invited to participate with their (semi-)automatic algorithms to accurately automated spinal curvature estimation and error correction from x-ray images. This paper aims to report results of the AASCE-2019 challenge[1]. At the time of writing this paper, more than 12 teams had submitted their results on the AASCE-2019 website[2]. In this paper, we focus only on those methods proposed by eight top-ranked teams.

The paper is arranged as follows. We first present the challenge dataset, rules for evaluation, and the established validation framework in Section 2. Each submitted method and its implementations will be summarized in Section 4. The validation results and their discussion of each method will be presented in Section 5, followed by conclusion in Section 6.

2. Data

There are in total of 707 spinal anterior-posterior x-ray images for training and testing collected from London Health Sciences Center in Canada using EOS medical imaging system. The IRB approval of our data has been obtained together with our two previous publications (Wang et al., 2019; Chen et al., 2019). The training and testing datasets consist of 609 and 98 x-ray im-

[1] https://aasce19.grand-challenge.org/Home/

[2] https://aasce19.grand-challenge.org/Home/

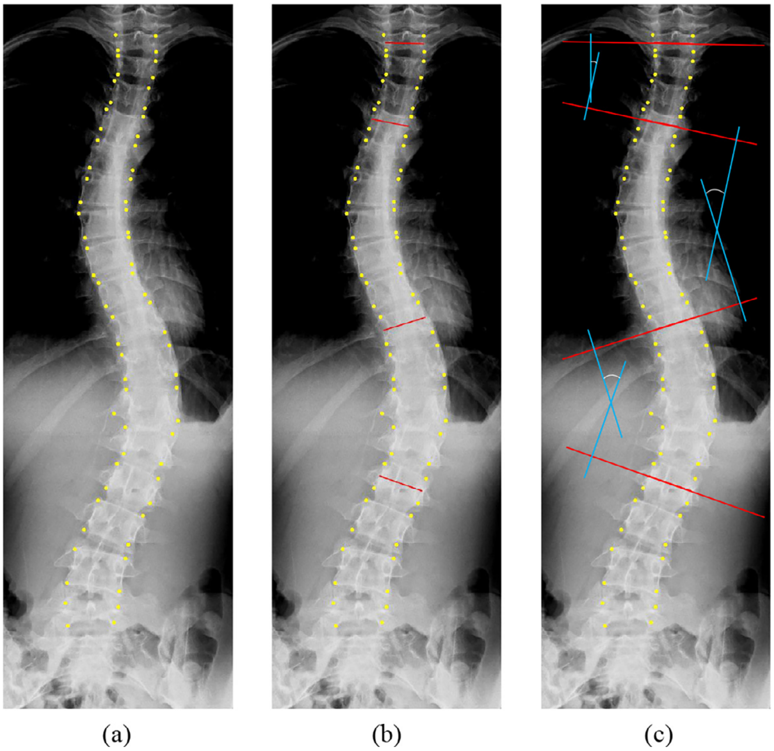

Fig. 3. Examples use landmarks (yellow points in subfigure (a), (b), and (c)) to measure Cobb angles. The most tilted vertebral endplates selected are drawn in red lines. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

ages with landmarks provided by two professional doctors in London Health Sciences Center and are available on the SpineWeb[3] Dataset 16). Since the cervical vertebrae are seldom involved in spinal deformity (O'Brien et al., 2008), 17 vertebrae consisting of the thoracic and lumbar vertebrae were selected as part of the spinal curvature evaluation. Each vertebra is marked by four landmarks with respect to four corners resulting in 68 points per spinal image. The Cobb angles for training were calculated using these landmarks. Given the landmarks, the code for calculating the Cobb angle based on landmarks was implemented by Matlab and is available on http://spineweb.digitalimaginggroup.ca/. Fig. 3 shows an example of spine images with landmarks and the most tilted vertebral endplates selected by the code. For any 4 points $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)$, form two vectors A and B, $A = (x_2 - x_1, y_2 - y_1), B = (x_4 - x_3, y_4 - y_3)$. The angle of the two vectors can be calculated as:

$$\alpha = arccos(\frac{A \cdot B}{||A||||B||}) \tag{1}$$

This code calculates the pairwise vectors' angles at all points using Eq. (1), finds the largest angle value and its corresponding endplates, and then finds the other two angles above and below. The details of this dataset are given in Table 2.

3. Evaluation

To evaluate the performance of different methods, we use the following metrics: symmetric mean absolute percentage (SMAPE), a circular mean absolute error (CMAE), Euclidean distance (ED), Manhattan distance (MD), and Chebyshev distance (CD).

Table 2
Statistics of AASCE dataset.

Specification	Train	Test
Number of images	609	98
Physical units (Dots Per Inch, DPI)	72	72
Pixels dimensions (SI units, micron)	144	144
Length of images (pixel) [min-max]	[973-3755]	[985-7316]
Width of images (pixel) [min-max]	[355-1427]	[273-2748]
Value of the Cobb angle (°) [min-max]	[0-156.39]	[0-60.93]
Number of not scoliosis (<10°)	20	11
Number of mild scoliosis (10° - 30°)	190	58
Number of moderate scoliosis (30° - 45°)	205	26
Number of severe scoliosis (>45°)	194	3
Number of some steel nails	53	4

The SMAPE metric is defined as Eq. (2):

$$SMAPE = \frac{1}{N}\sum_{i=1}^{N}\frac{SUM|X_i - Y_i|}{SUM(X_i + Y_i)} \times 100\% \tag{2}$$

where X_i is the estimated Cobb angles, Y_i is the ground truth, $X_i = (\alpha_{i1}, \alpha_{i2}, \alpha_{i3})$ and $Y_i = (\beta_{i1}, \beta_{i2}, \beta_{i3})$, N is the number images (the smaller, the better).

The CMAE metric is defined as Eq. (3):

$$\begin{aligned} CMAE &= \tfrac{1}{N}\textstyle\sum_{i=1}^{N} CMEAN(X_i - Y_i) \\ X_i - Y_i &= (\alpha_{i1} - \beta_{i1}, \alpha_{i2} - \beta_{i2}, \alpha_{i3} - \beta_{i3}) = (\theta_1, \theta_2, \theta_3) \\ CMEAN(\theta_1, \theta_2, \theta_3) &= arctan(\tfrac{\bar{y}}{\bar{x}}) \\ \bar{x} &= \tfrac{1}{3}(cos(\theta_1) + cos(\theta_2) + cos(\theta_3)) \\ \bar{y} &= \tfrac{1}{3}(sin(\theta_1) + sin(\theta_2) + sin(\theta_3)) \end{aligned} \tag{3}$$

where X_i is the estimated Cobb angles, Y_i is the ground truth, $X_i = (\alpha_{i1}, \alpha_{i2}, \alpha_{i3})$ and $Y_i = (\beta_{i1}, \beta_{i2}, \beta_{i3})$, N is the number images (the smaller, the better).

[3] http://spineweb.digitalimaginggroup.ca/

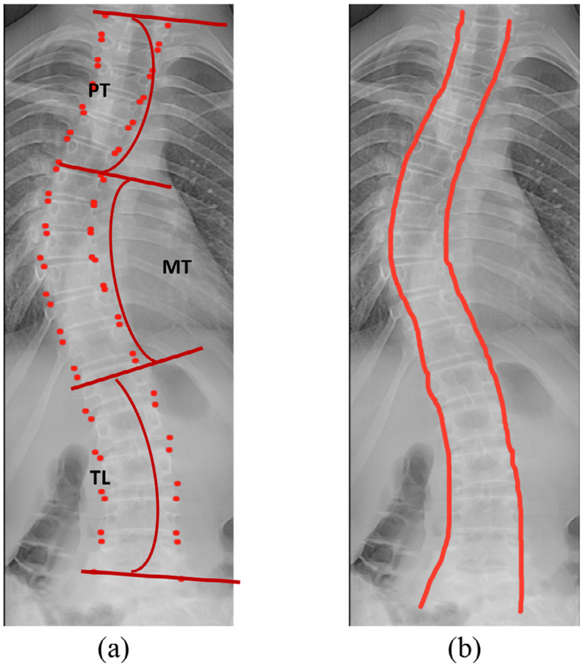

Fig. 4. (a) Cobb angles and hand-crafted landmarks of the spine, (b) continuous boundary of the spine.

The Euclidean distance is defined as Eq. (4):

$$ED = \frac{1}{N}\sum_{i=1}^{N}\sqrt{(\alpha_{i1}-\beta_{i1})^2+(\alpha_{i2}-\beta_{i2})^2+(\alpha_{i3}-\beta_{i3})^2} \tag{4}$$

where $(\alpha_{i1}, \alpha_{i2}, \alpha_{i3})$ is the estimated Cobb angles, $(\beta_{i1}, \beta_{i2}, \beta_{i3})$ is the ground truth, N is the number images (the smaller, the better)

The Manhattan distance is defined as Eq. (5):

$$MD = \frac{1}{N}\sum_{i=1}^{N}(|\alpha_{i1}-\beta_{i1}|+|\alpha_{i2}-\beta_{i2}|+|\alpha_{i3}-\beta_{i3}|) \tag{5}$$

where $(\alpha_{i1}, \alpha_{i2}, \alpha_{i3})$ is the estimated Cobb angles, $(\beta_{i1}, \beta_{i2}, \beta_{i3})$ is the ground truth, N is the number images (the smaller, the better)

The Chebyshev distance is defined as Eq. (6):

$$CD = \frac{1}{N}\sum_{i=1}^{N}max(|\alpha_{i1}-\beta_{i1}|, |\alpha_{i2}-\beta_{i2}|, |\alpha_{i3}-\beta_{i3}|) \tag{6}$$

where $(\alpha_{i1}, \alpha_{i2}, \alpha_{i3})$ is the estimated Cobb angles, $(\beta_{i1}, \beta_{i2}, \beta_{i3})$ is the ground truth, N is the number images (the smaller, the better)

4. Methods and implementations

A total of more than 12 teams successfully submitted their results to AASCE2019 before the official deadline. Here we analyze the results of eight top ranked teams.

4.1. XMU: Xiamen University

Intuition [4] Team XMU extended the landmarks to the continuous spine boundary (shown in Fig. 4(b)) to introduce attention information for better feature extraction, and they used a convolutional neural network to regress angles directly.

Method The proposed algorithm workflow is shown in Fig. 5. The workflow consists of two stages: 1) the boundary segmentation network and 2) the angle regression network. The boundary segmentation network takes an X-ray image as input and generates a spine boundary prediction map. The segmentation network applies a symmetrical encoder-decoder structure and consists of lots of residual blocks. Inspired by U-Net (Ronneberger et al., 2015), there are two long-distance skip connections at different resolution levels to aggregate the shallow and deep features. At the bottom of the network, the Pyramid Pooling Block (PP block) (Zhao et al., 2017) is used to collect global information. The angle regression network takes the X-ray image and the spine boundary prediction map as inputs to predict the Cobb angle. Due to the effect of Dense Net (Huang et al., 2017), DenseNet121 is selected as the backbone of the regression network and directly predicts three Cobb angles at the last layer.

Implementation In the implementation, Team XMU connected all landmark points in an image and refined the two coarse boundaries by dilation operation to obtain ground truths for the boundary segmentation network. Team XMU trained the segmentation and regression networks independently and applied the Dice loss and MSE loss as supervision in two supervised tasks, respectively. Moreover, two networks are both optimized by the Adam optimizer with the learning rate of 1 × 10^{-4}.

[4] https://github.com/wangshuxinxinxin/SCG-Net

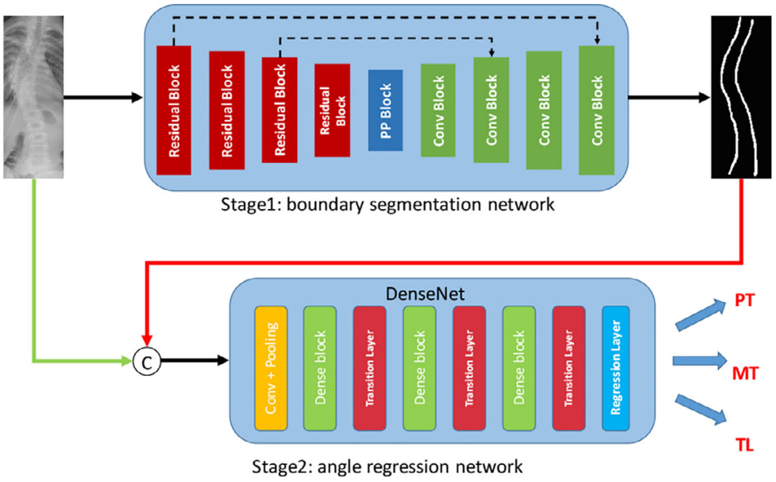

Fig. 5. The workflow of Team XMU's method. In stage 1, the boundary segmentation network generates the spine boundary prediction map from X-ray images. In stage 2, the angle regression network combines the X-ray image with the previous spine boundary prediction map to predict the Cobb angles.

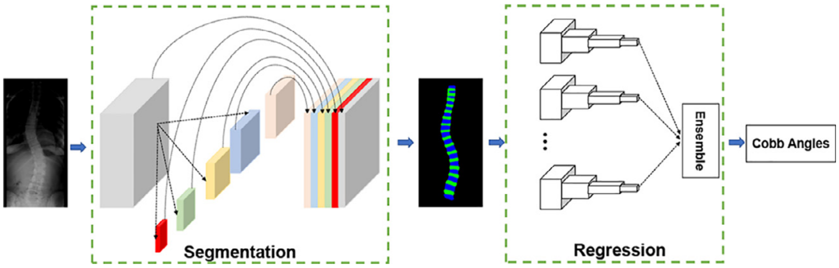

Fig. 6. The framework proposed by Team Tencent for automated spine curvature estimation.

4.2. Tencent: Tencent YouTu Lab

Intuition [5] Team Tencent regarded the vertebrae and intervertebral space segmentation as an intermediate state and ensembles multiple networks to produce angles.

Method Their proposed framework (shown in Fig. 6) consists of two subnets: one for segmentation and the other for regression. The first stage modifies PSPNet (Zhao et al., 2017) to segment the vertebrae and intervertebral space. The ResNet-101 (He et al., 2016) is used as a feature extractor to leverage high-level convolutional features and the global pyramid pooling layer is used to combine the local and global clues. Furthermore, the dilated convolution is appended with 2, 4, and 6 dilation rates to further capture various receptive fields. The objective function of the segmentation task is the combination of cross-entropy loss and the dice coefficient loss. In the second stage, Team Tencent utilized recent popular networks, including ResNet (He et al., 2016) series, DenseNet (Huang et al., 2017) series and EfficientNet (Tan and Le, 2019) series to perform the regression task, and all of the results are combined to reduce generalization error. The objective function is defined as Eq. (7):

$$\mathcal{L}(X, Y, \theta) = \sum_{i}^{c} \frac{|Y_i - F(X) + \epsilon|}{|Y_i + F(X) + \epsilon|} + \lambda \sum_{i}^{k} |\theta_i| \tag{7}$$

where c is the number of angles, X is the input segmentation mask, Y is the ground-truth mask, $F(X)$ is the predicted angles, and θ is the set of model parameters. In addition, considering the domain shift between the training and testing distributions, they follow the instruction of Ganin and Lempitsky (2015) for domain adaptation with a gradient reversal layer.

Implementation Team Tencent pre-processed the X-ray data before inputting it to their network. First, they apply histogram equalization to ease the domain shift problem. Then, they conduct augmentations for training data, including random rescaling between $[0.85, 1.25]$ and random rotation between $[-45^\circ, 45^\circ]$. Team Tencent tried to add Gaussian noise to the input images, but it did not help. The whole network was randomly initialized from a Gaussian distribution $N(0, 0.01)$, and the learning rate was initialized as 3×10^{-3} with cosine decay schedule. They used Adam optimizer where the λ and β are set to 0.9 and 0.999, respectively. The input was resized to 1024×512 and 512×256 for segmentation and regression network, respectively. Team Tencent implemented their method in the PyTorch framework using 4 NVIDIA P40 GPUs.

4.3. iFLYTEK: iFLYTEK research South China, computer vision group

Intuition [6] Team iFLYTEK only focused on landmarks detection, and the Cobb angles measurement is handed over to the Matlab code provided. As the spinal keypoint detection can be divided into rough vertebra/keypoint detection and fine keypoint detection, Team iFLYTEK introduced traditional detection methods RetinaNet (Lin et al., 2017) to vertebra detection as Method-1 and Simple Baseline (Xiao et al., 2018) and HR-Net (Sun et al., 2019) to spinal keypoint detection as Method-2.

Method As shown in Fig. 7, their method can be seen as two parallel processes. Method-1 can work for the vertebrae's unfixed number, but difficult to model the scene sequentiality and hard to balance the precision and recall. Method-2 can grasp the global implicit sequentiality of keypoints through generating corresponding heatmaps simultaneously with a fixed order.

In Method-1, RetinaNet (Lin et al., 2017) is provided to detect individual vertebrae in a spine and generate corresponding bounding boxes, while HR-Net (Sun et al., 2019) is designed to detect 4 key points in a bounding box. For RetinaNet, they follow (Lin et al., 2017) to experiment with ResNet-50-FPN backbone. The model is trained for 180k iterations with a total of 8 images per minibatch. Following (Sun et al., 2019), they use HRNet-W48 with an input size of 384×288, and the training process is terminated within 30 epochs.

Method-2, which follows the workflow shown in Fig. 7, is robust and adequate for diverse shapes of the spine. Firstly Simple Baseline I is trained with spinal images to detect all 68 key points directly. The predicted keypoints can smoothly trace the curvature of most spines. Thus, the predicted keypoints are used as the outline sketches of spines to generate patches.

Implementation A patch includes n points, $\{n|n = 4, 6, 8, 10, 12\}$, for 1 to 3 vertebrae. Half of the vertebra is allowed, but there must be at least two halves of the vertebra in a patch. Patches are randomly captured multiple times within a certain vertebrae range. Such Patch Process not only makes it much easier to match a template for any adjacent vertebrae but also makes it easy to augment a large amount of data, so that the following Simple Baseline II becomes a highly robust model. The uncertain number of keypoints detected by Simple Baseline II are as shown in Fig. 7(f). Adding a PostProcess helps remove outliers, handle the squeezed vertebrae, and cluster the final 68 key points from 4 vertex groups by DBSSCAN and K-means. Following (Xiao et al., 2018), the ResNet-152 is utilized as the encoder network in both Simple Baselines I and II, with CLAHE enhancement. For Simple Baseline I, the training ends within 200 epochs with a batch size of 32 and the same input size as HR-Net. As for Simple Baseline II, the input size is 256×192 with 130 images per batch, and the training lasts for 70 epochs.

4.4. XDU: school of electronic engineering, Xidian University

Intuition [7] Team XDU proposed an algorithm for accurate landmark detection, and Cobb angles are produced by Matlab code provided with landmark coordinates. The method followed the coarse-

[5] https://github.com/hust-linyi/Seg4Reg

[6] https://github.com/YJY-CV/Spine

[7] https://github.com/zzs95/AASCE2019_code

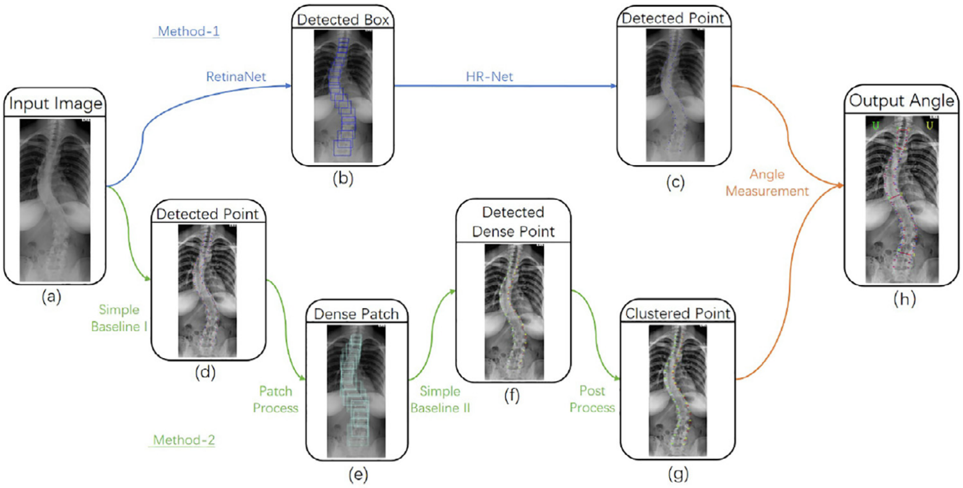

Fig. 7. Illustration of Team iFLYTEK's parallel framework that propagates two proposed methods. (a) is the input image, the blue bounding boxes in (b) and key points in (c) are prediction from RetinaNet and HR-Net in Method-1. The bottom workflow shows the Method-2, including (d) rough points generation, (e) patch selection, (f) patch points generation and (g) clustering. (h) is the final image that shows the process of spine curvature estimation, with the red and blue keypoints respectively representing ground truth and prediction. The lines show which vertebrae are selected to calculate the Cobb angles. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

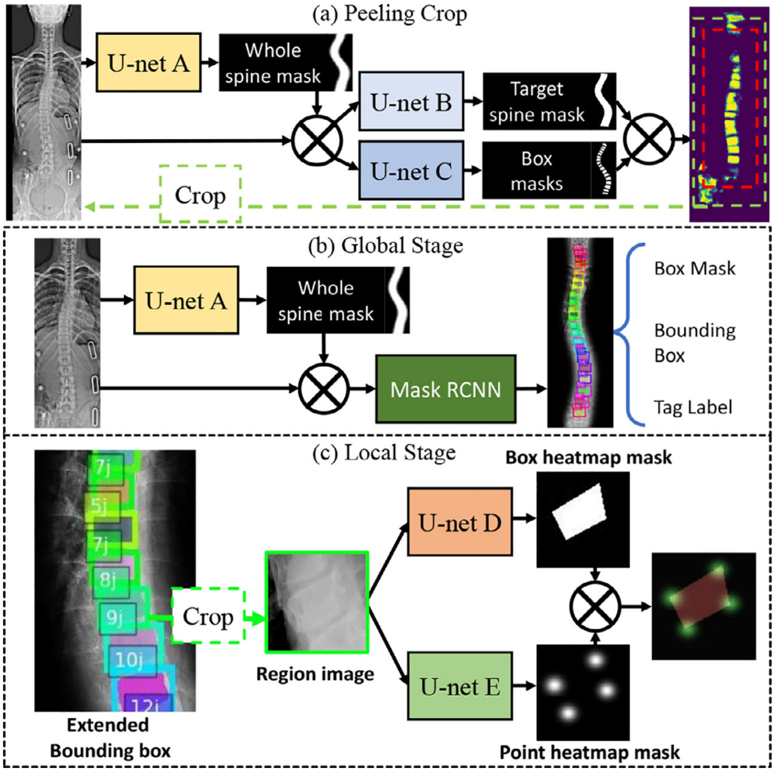

Fig. 8. Team XDU's framework. Peeling Crop is the data pre-process shown in (a). A schematic illustration of the proposed 2-stage regression method, consisting of (b) the Mask RCNN in the global stage and (c) the multi-channels heatmaps regression in the local stage.

to-fine pipeline, which can inform the global locations for vertebrae and then obtain the 4 corner coordinates for each vertebra in the local stage.

Method The framework is shown in Fig. 8, including pre-process stage Peeling Corp, Global Stage and Local Stage. Peeling Corp is progressive, gradually narrowing the unwanted margins, similarly to peeling onion piece by piece. Three trained U-Net (Ronneberger et al., 2015) models regress 3 kinds of heatmaps, which are Whole Spine Mask (WSM), Target Spine Mask (TSM), and Box Masks. The WSM and TSM indicate the location of the spine to filter out the false-positive regions while inferring. The first 17 channels of Box Masks highlight separately for the vertebrae. The input image multiplies with WSM, to regress TSM and Box Masks. To obtain the main target spine region, the TSM multiplies with the first 17 channels of the Box Masks. Each channel obtains the highlight location. Based on these two endplates location, the detected region box (the red dotted box) is extended with a fixed height-width ratio in the over cropping in the target spine. In the extended box (the cyan dotted box), the next iteration crops the input image. The Peeling Crop iterates the cropping and the predicting and stops with the minimal narrowing distance and the maximum iteration numbers. The Peeling Crop processes

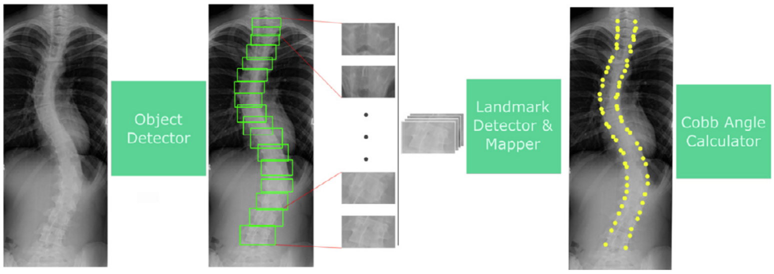

Fig. 9. Team NAAMII's two-step framework: An object detector first detects the 17 vertebrae from the input images. These bounding box patches are fed separately to a landmark detector that regresses the four corners of the vertebra. Cobb angles are calculated using the landmarks that are mapped back to the original image. Faster-RCNN and DenseNet are used for object detection and landmark detection respectively.

the testing data once, before the inferring in the global and local stage.

In the global stage, Mask R-CNN is used as backbone. The preprocessed images are inputted to U-net A, then multiply with the WSM and inputted to the backbone model. The trained Mask R-CNN (He et al., 2017) predicts the box masks, bounding boxes and tag labels. Then the model extracts global information for guiding the local stage detection. The local stage obtains the landmarks for each vertebra in the coarse bounding box. As shown in Fig. 8(b), the local stage consists of U-net D and U-net E. These two models are trained with the region images to regress the box heatmaps and the point heatmaps. The first 4 channels of the point heatmaps with 2D Gaussian distributions at the center of the landmark are masked separately by the box heatmaps. The overlapping region of the first 4 channels contains the locations of the landmarks. Each coordinate is obtained as the mean position of the pixels whose values are greater than the threshold of 0.5. The 4 coordinates of 1 vertebra are obtained in the first 4 overlapping channels. The local stage procedure traverses the bounding boxes and obtains the whole spine landmarks. At the post-processing, the 2 sets of coordinates from bounding boxes and box masks are used to fit the spine curve to finetune these error predictions, filter out error predictions far from the curve and rotate the coordinate pairs on their intersections closing to the vertical lines.

Implementation The proposed method was implemented in Python on a desktop with 4.0 GHz Intel@i7 CPU and a TITAN X Pascal graphics card with 12 GB GPU memory. The U-net models were implemented in PyTorch, and reused in the framework. The models regress an additional shared background channel to handle the class-imbalance problem. The Mask RCNN was implemented in MXNet with the default settings. The Whole Spine Mask, the Target Spine Mask, the Box Masks, and the bounding boxes were derived from the coordinates data. The models were trained separately, and the modular framework is easy to implement.

4.5. NAAMII: NepAl applied mathematics and informatics institute for research

Intuition [8] Team NAAMII also presented a direct landmark detection algorithm. Cobb angles were calculated using the MATLAB code provided. The difficulty in vertebra landmarks compared to other anatomical landmarks is the presence of a large number of similar-looking vertebrae.

Method Team NAAMII proposed a two-step novel framework consisting of two separately trained networks for a) vertebra detection followed by b) landmarks detection. Fig. 9 shows the overall pipeline. In each image, the 17 vertebrae are detected using a bounding box object detector. Each of the predicted bounding box patches is then fed as individual images to a landmark detector network. The four corner landmarks predicted for each bounding box patch are mapped back to the original image, generating 68 landmarks per image. The outliers are removed from the predicted landmarks using post-processing techniques, after which the three Cobb angles are calculated.

Implementation To create GT bounding boxes for training object detector, Team NAAMII connected the four landmark corners of each vertebra, creating a box whose width and height were then increased symmetrically by 50 and 10 pixels respectively. Faster-RCNN (Ren et al., 2015), a widely used two-stage object detector, is used for object detection. The base network used for Faster-RCNN is ResNet V1 101 with pre-trained weights on Imagenet data, which was fine-tuned after block 2. Since the vertebrae are relatively small and do not have extreme aspect ratios, the selected scale and aspect ratio for the anchor boxes of Faster-RCNN were: the scale of 64^2 and 128^2 pixels and aspect ratios of $1:1$ and $2:1$. The other training details included training of around 180k steps, batch size 1, SGD optimizer with momentum 0.9, learning rate 0.0003, and early stopping. The implementation was adopted from Luminoth[9] in Tensorflow framework 10.1. Data augmentation included random Gaussian noise ($\mu = 0$, $\sigma = 0.005$), and vertical and horizontal flips with a probability of 0.5. Rescaling of the images was done preserving the aspect ratio such that its sizes remained within 600–1000 pixels as much as possible.

Densely Connected Convolutional Neural Network (DenseNet) (Huang et al., 2017) are used for landmark regression. Team NAAMII used 5 blocks with a growth rate of 8, where the growth rate is the number of output feature maps of each layer. After 5 blocks, a 2D Global Average Pooling is used, which is followed by a dense layer. The linear activation function is used in the final layer consisting of 8 output units. All the input images were resized to 200×120 pixels. Some falsely detected vertebrae away from spine curvature were rejected as outliers. They also smoothed the spine curvature with the order 6 polynomial fit on detected landmarks.

4.6. UC: University of California, San Francisco

Intuition [10] Team UC proposed an end-to-end network to predict Cobb angles, and used an additional dataset Chest X-rays (Irvin et al., 2019). Their approach estimates Cobb angles from spine radiographs via a data-efficient method based on transfer learning, wherein the final model is fine-tuned over a pre-trained checkpoint from another closely related domain.

Method Fig. 10 illustrates the three stages of training; first, pretraining on a much larger chest x-ray dataset, followed by two stages of fine-tuning - for landmark detection and Cobb angle estimation, respectively.

[8] https://github.com/Bidur-Khanal/SpineCurvEst

[9] https://github.com/tryolabs/luminoth

[10] https://github.com/radiology-guru/cobb-angles/tree/main/cobb_angles

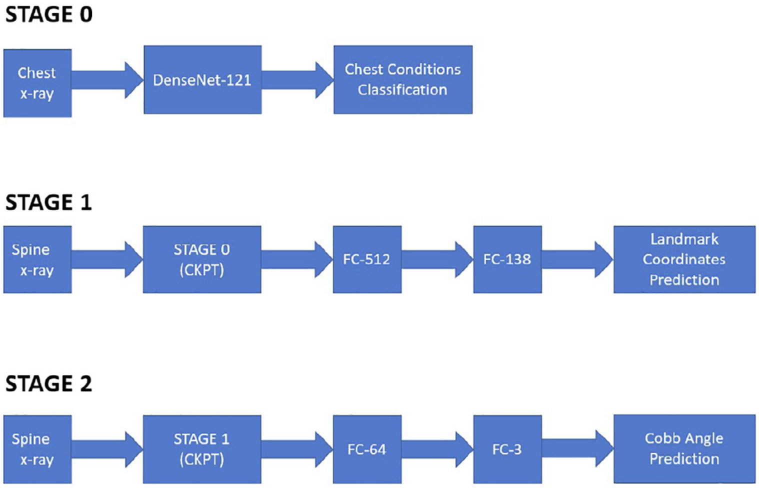

Fig. 10. Team UC's framework.

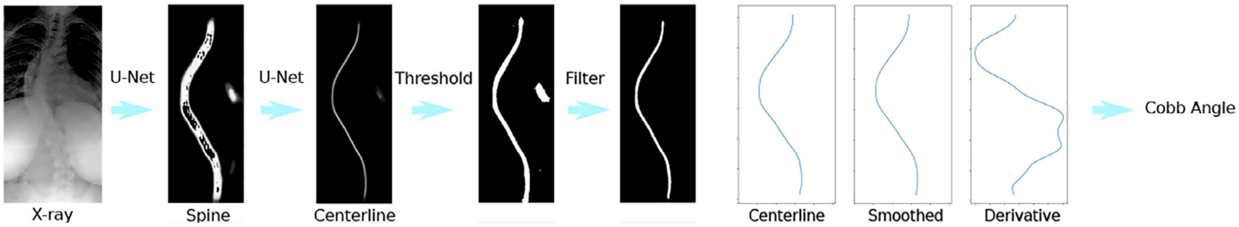

Fig. 11. Team ErasmusMC's framework. Left to right: input X-ray scan (*X-ray*); spine and centerline segmentation with cascaded U-Net networks; thresholding of the centerline segmentation; removing small connected components; extraction of the spine centerline curve (*Centerline*); centerline smoothing using heat equation (*Smoothed*); computing the derivative of the centerline (*Derivative*), computation of Cobb angles.

The following approach is adopted for detecting vertebral landmarks: 1) A CNN feature extractor (DenseNet-121) is trained on the CheXpert dataset (Irvin et al., 2019). 2) Fully connected layers are added to the pre-trained DenseNet-121 model for predicting landmarks. 3) The new network is fine-tuned with labeled examples from the AASCE training dataset. Cobb angle estimation is a regression problem, with the three angles being the target values for prediction. An L1 loss is used along with the direct optimization of the SMAPE metric. The DenseNet-121 feature extractor from landmark detection is used as the initial checkpoint, which is then fine-tuned to predict the Cobb angles by introducing additional fully-connected layers in the DenseNet model.

Implementation The model was implemented entirely in PyTorch using torchvisions DenseNet-121 module and trained on a single NVIDIA K80 GPU. Standard data augmentation such as random flip, crop, and resize, were also used for training the models.

4.7. ErasmusMC: Department of radiology and nuclear medicine

Intuition [11] Team ErasmusMC used another form of cobb angle calculation and designed an end-to-end algorithm. Cobb angles were measured directly from the centerline of the spine, which was automatically segmented from X-ray scans using cascaded neural networks optimized end-to-end (Fig. 11).

Method Team ErasmusMC first extracted the centerline of the spine by using two cascaded U-Nets (Ronneberger et al., 2015). The architecture of each U-net has fewer feature maps and batch normalization layers (Ioffe and Szegedy, 2015) before each pooling layer. The X-ray scan is given as input to the first network. The first network is optimized to compute the segmentation of the spine, and the second network is optimized to compute the segmentation of the spine centerline alone. The networks are optimized end-to-end, simultaneously, using Adadelta optimizer (Zeiler, 2012). The loss function is the mean squared error over voxel intensities between the binary ground truth and predicted segmentation. Input X-ray scans are augmented at random on-the-fly during training, with rotation, translation, and horizontal flipping, the addition of Gaussian noise, and varying brightness and contrast.

To ensure continuity of the centerline, the centerline pseudo-probability map at the output the network – before the sigmoid activation – is binarized using a low threshold (0.25). This process also creates small connected components, which removed by filtering. Team ErasmusMC detected the two borders of the centerline segmentation and chose the points located halfway in between to model the centerline curve. Because of local noise due to either the low image resolution or errors in the segmentation, the derivative of the centerline can be substantially perturbated. To avoid this issue, Team ErasmusMC smoothed the centerline curve using the heat equation, solved with the Euler method. Team ErasmusMC set the heat transfer coefficient to 0.01, and the number of iterations in Euler's method set to 65000. They tuned the number of iterations on the training set.

Implementation Cobb angles are computed following the formula presented by Horng et al. (2019), and adapted to the Eq. (8):

$$\phi = \frac{180}{\pi}\left| arctan\left(\frac{T(p_{R,M}) - T(p_{R,m})}{1 + T(p_{R,M}) \cdot T(p_{R,m})}\right)\right|, \tag{8}$$

where $T(p)$ is the tangent slope of the centerline at point p, $p_{R,M}$ is the point with the maximum slope in the region R, and $p_{R,m}$ is the point with the minimum slope in R. To obtain the tangents, the derivative of the centerline in every point is first computed ("Derivative" in Fig. 11). Then, values of the slope of the tangent are

[11] https://github.com/fpgdubost/aasce_emc

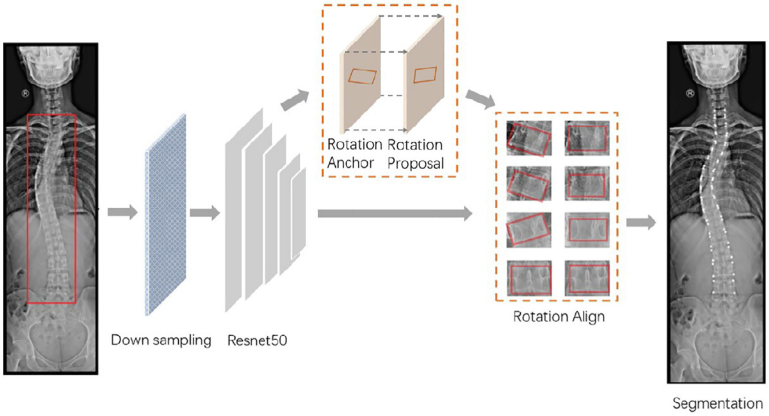

Fig. 12. The pipeline of the proposed method. Team PAT uses a combination of ResNet50 and FPN as the backbone for multiscale feature extraction.

considered, $T(p)$, only in 19 points evenly spaced over the centerline as an approximation of the location of the interstices between the 17 annotated vertebrae. The entire centerline is used as the region of interest R in the equation above to compute the major angle. The upper and lower angles are computed in the regions above and below the highest curvature region, defined as the region between $p_{R,m}$ and $p_{R,M}$.

To improve generalization performance, ensembles of networks are used. Those networks are optimized on different random subsets of the training set, and their predictions are averaged at two stages of the pipeline: 1) centerline segmentation maps of several models are averaged pixel-wise, before the computation of Cobb angles, and 2) Cobb angles predicted by several ensembles are also averaged.

4.8. PAT: PingAn technology

Intuition [12] Team PAT proposed a two-stage automated spinal landmarks detection network based on rotational regional proposals of vertebrates. Their method addresses the problem that horizontal rectangular vertebrate proposals cannot be suitable for severe spinal curvature estimation, since vertebrates are heavily titled, and accuracy of both two sub-tasks can be affected due to vertebrate proposal misalignment. Cobb angles calculation is given to the provided Matlab code.

Method In stage one, the proposed network detects the location of vertebrates using rotated rectangular regions. In stage two, each vertebrate region undergoes rotation co-registration using a rotation angle from the previous stage. Landmarks detection is then performed on aligned proposals. Finally, they estimate the Cobb angle using detected vertebrate landmarks.

The backbone of their approach is the feature pyramid network (FPN) (see Fig. 12). FPN performs multi-scale feature extraction and gives regional proposals on each scale. For training, the ground truth of each rectangular vertebrate bounding box is defined as 5 parameters (x, y, w, h, θ). Especially, (x, y) describes the center location of the bounding box, and (w, h) describes the dimension of the bounding box. The rotation angle θ is about the angle of the inclined bounding box with respect to the x-axis, and the rotation center is fixed at (x, y), ranging from $-\frac{\pi}{2}$ to $\frac{\pi}{2}$. Besides, rotational anchors are designed at 5 scales, 3 ratios, and 3 rotational angles. The rotation align is shown in Fig. 12. After obtaining 5 parameters (x, y, w, h, θ), rotated vertebrate proposals are adjusted using θ. Then region of interest (ROI) alignment is adopted to the proposals for getting a fix-size feature map as stated in (He et al., 2017). Finally, a fully convolutional network (FCN) are used for landmarks segmentation.

Table 3
Source codes from top-ranked teams.

Team	Link
XMU	https://github.com/wangshuxinxinxin/SCG-Net
Tencent	https://github.com/hust-linyi/Seg4Reg
iFLYTEK	https://github.com/YJY-CV/Spine
XDU	https://github.com/zzs95/AASCE2019code
NAAMII	https://github.com/Bidur-Khanal/SpineCurvEst

To reduce the impact of dataset differences, Team PAT proposes a standardized image preprocessing program. First, in order to remove the head and pelvis floor, Team PAT uses horizontal intensity projection to separate the spine region. The statistical evaluation of the entire data set determines the intensity thresholds for different body parts. Second, after obtaining the spine region, they perform spinal cord segmentation and midline extraction for spinal cord co-registration and false-positive key points suppression. Finally, the spine area is further refined with an aspect ratio of 3.5 locked between the image height and width.

Implementation The optimization of network parameters is performed using Adam optimizer with a learning rate of 1×10^{-4}. For stage one, Team PAT use minimum mean square error loss for rotation angle regression, a smooth L1 loss for bounding box regession. For stage two, Team PAT use binary cross entropy loss combined with dice loss for landmarks segmentation.

5. Discussion

Open code is necessary and fair for reproducibility. We have convinced all teams to make their code open. By far, we have five teams provided their GitHub code, as shown in Table 3, including the ranking #1 and #2 teams. We believe these codes will give key information on how the challenge is solved and implemented for readers.

Based on Section 4, all mentioned teams employed deep learning based algorithms. Morphological information of the spine is considered in their methods. Therefore, these algorithms can be summarized into two categories: the first category detecting the region of interest (ROI) and the second category estimating the Cobb angle. For ROI detecting, three methods (Team iFLYTEK, Team XDU, Team NAAMII and Team PAT) used vertebra detection (with Faster R-CNN, Mask R-CNN, or FPN) followed by landmarks detection. Two methods (Team XDU and Team UC) estimate the spine

[12] https://github.com/PASpine/Xspine

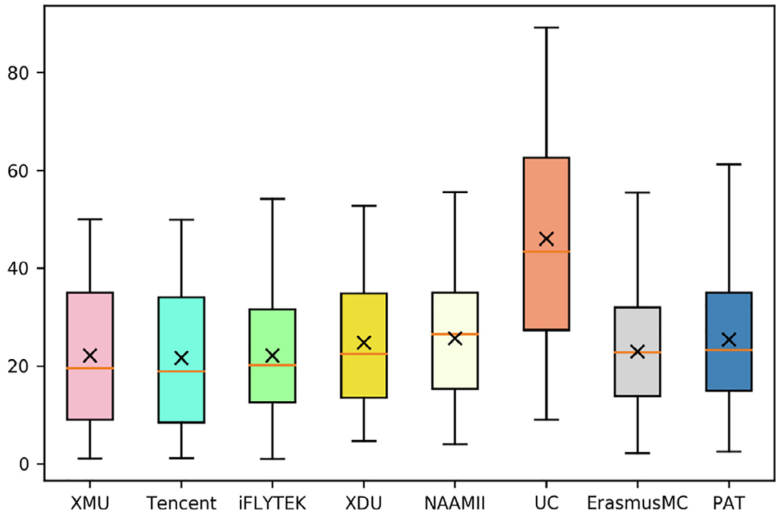

Fig. 13. SMAPE boxplot of spine images. The cross represents the average values of the scores. Red lines inside the boxes represent medians. Separate dots show scores of each case.

Table 4
Average performance of each team on five metrics. Symbol (1) means the best score under the metric, Symbol (2) means the second-best score under the metric.

Team	SMAPE (%)	CMAE (°)	ED (°)	MD (°)	CD (°)
Tencent	21.71 (1)	4.85 (1)	11.17 (1)	14.55 (1)	10.16 (1)
XMU	22.18	4.91 (2)	11.23 (2)	14.74 (2)	10.17 (2)
ErasmusMC	22.96	5.69	12.12	17.07	10.52
iFLYTEK	22.17 (2)	5.48	12.14	16.45	10.74
XDU	24.8	6.28	13.18	18.83	11.27
NAAMII	25.7	6.69	14.14	20.08	12.32
PAT	25.48	6.58	14.37	19.76	12.94
UC	45.99	17.13	32.01	51.41	24.09

curve (boundary or centerline). The rest of all methods directly estimate landmark coordinates or get a spine segmentation map firstly. For Cobb angle estimating, four teams (Team iFLYTEK, Team XDU, Team NAAMII and Team PAT) directly used the Matlab code we provided to calculate the Cobb angle by landmarks coordinates. The other three teams regard it as a regression problem, with the three angles directly estimated. One team (Team ErasmusMC) uses the tangent slope of the centerline to calculate the Cobb angle, as shown in Eq. (8). This form of calculating the Cobb angle can get the same results as the clinical Cobb angle calculation, but in a differentiable form.

In this section, we provide more analysis for each team in different scoliosis levels and insightful discussions to inspire readers.

5.1. Evaluation in terms of all scoliosis ranges

We first evaluate the performance of all scoliosis ranges. Table 4 shows the average performance of each team with five metrics. It can be seen that Team Tencent is the best in all metrics, and Team XMU ranks second-best in the four metrics. Both teams use an end-to-end network and consider the Cobb angle calculation as a regression problem. Figs. 13 and 14 show boxplots of in total of 98 images of each method on SMAPE and CMAE, respectively. Besides medians, the means are also indicated by the black cross. Team Tencent performed best overall as shown in boxplots, but it is more scattered than Team iFLYTEK. In Fig. 13, first 4/5 teams have pretty similar results because they used similar network architecture and also considered the Cobb angle calculation as a regression problem that can reflect the scoliosis deformation parameters. Therefore, they get similar training results. Furthermore, these teams use the end-to-end networks in which all parameters of the model can be simultaneously trained for one loss function, and it turned out to be very effective in vertebra detection and segmentation. It can be clearly seen from Fig. 14 that all teams have some outliers, that is, cases of failure. For further analysis, we calculated Wilcoxon signed-rank test, as shown in Tables 5 and 6. Interestingly, we did not find any method of achieving robust and statistically significant better performances compared to all other methods. For example, we found that the results from Team Tencent present the highest mean in terms of SMAPE and CMAE. Nevertheless, their differences with the results obtained from Team ErasmusMC are not strongly statistically significant. To better reflect the accuracy of the eight methods, we further measured the correlation coefficients of the max angle. Correlation coefficients are used to measure the strength of the relationship between the ground truth and the angle predicted by the proposed methods. A correlation coefficient with a value of 0.9 or higher would represent a very strong positive relationship. As shown in Fig. 15, except for Team UC, the other methods have obtained values greater than 0.9. It shows that these methods achieve a high correlation coefficient with manual measurement. In summary, no team can rank first in all evaluation metrics, suggesting that we can further choose or design evaluation metrics according to clinics in the future study.

5.2. Evaluation in terms of not scoliosis and mild scoliosis

Besides evaluation in terms of all scoliosis range, we further evaluate the performance based on not scoliosis and mild scoliosis. Those with Cobb angle of less than 10° usually think that there is no scoliosis. A total of 11 subjects have no scoliosis. As shown in Fig. 18 (see Appendix 1), we can see that Team ErasmusMC achieved a correlation coefficient with the highest value of 0.70, and Team NAAMII got the second-highest value of 0.62. It shows that Team ErasmusMC's method is more sensitive to very mild scoliosis, which may be caused by their centerline curve prediction and the way Cobb angle calculated. Team NAAMII can get a more accurate landmark prediction for scoliosis that is not so deformed. The performances based mild scoliosis is shown in Fig. 19 (see Appendix 1). Team XMU and Team Tencent got a similar highest value of 0.81, and their method performed best with Cobb angle between 10 and 30 degrees. In addition to Team UC, other teams also got a good value in this range.

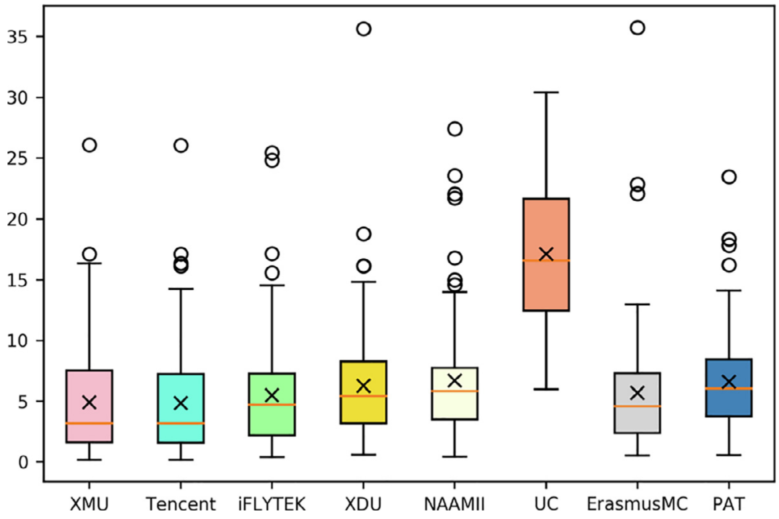

Fig. 14. CMAE boxplot of spine images. The cross represents the average values of the scores. Red lines inside the boxes represent medians. Separate dots show scores of each case.

Table 5
The p-values by performing the Wilcoxon signed-rank test on SMAPE of any two teams among the eight teams. Single underline denotes weak statistical significance (p-value < 0.05). Double underline denotes strong statistical significance (p-value < 0.01).

	Tencent	iFLYTEK	XDU	NAAMII	UC	Erasm-usMC	PAT
XMU	4.69×10^{-1}	5.36×10^{-1}	1.10×10^{-1}	1.14×10^{-1}	1.70×10^{-12}	5.62×10^{-1}	1.28×10^{-1}
Tencent	–	4.26×10^{-1}	6.99×10^{-2}	7.32×10^{-2}	1.62×10^{-12}	4.28×10^{-1}	8.41×10^{-2}
iFLYTEK	–	–	2.57×10^{-2}	1.15×10^{-3}	2.21×10^{-15}	1.47×10^{-1}	5.48×10^{-5}
XDU	–	–	–	2.93×10^{-1}	2.46×10^{-14}	5.78×10^{-2}	2.92×10^{-1}
NAAMII	–	–	–	–	1.91×10^{-13}	1.04×10^{-2}	6.69×10^{-1}
UC	–	–	–	–	–	2.15×10^{-15}	7.00×10^{-16}
Erasm-usMC	–	–	–	–	–	–	1.87×10^{-3}

Table 6
The p-values by performing the Wilcoxon signed-rank test on CMAE of any two teams among the eight teams.single underline denotes weak statistical significance (p-value < 0.05). Double underline denotes strong statistical significance (p-value < 0.01).

	Tencent	iFLYTEK	XDU	NAAMII	UC	Erasm-usMC	PAT
XMU	$\underline{9.74}\times10^{-3}$	5.46×10^{-2}	$\underline{7.12}\times10^{-3}$	$\underline{2.38}\times10^{-3}$	$\underline{3.48}\times10^{-16}$	9.83×10^{-2}	$\underline{1.16}\times10^{-3}$
Tencent	–	$\underline{4.14}\times10^{-2}$	$\underline{3.21}\times10^{-3}$	$\underline{1.62}\times10^{-3}$	$\underline{3.48}\times10^{-16}$	7.39×10^{-2}	$\underline{7.00}\times10^{-4}$
iFLYTEK	–	–	$\underline{8.87}\times10^{-3}$	$\underline{1.16}\times10^{-4}$	$\underline{3.18}\times10^{-16}$	6.44×10^{-1}	$\underline{2.18}\times10^{-6}$
XDU	–	–	–	3.41×10^{-1}	$\underline{1.71}\times10^{-15}$	$\underline{8.78}\times10^{-3}$	1.52×10^{-1}
NAAMII	–	–	–	–	$\underline{1.54}\times10^{-14}$	$\underline{6.40}\times10^{-4}$	5.86×10^{-1}
UC	–	–	–	–	–	$\underline{4.14}\times10^{-16}$	$\underline{3.91}\times10^{-16}$
Erasm-usMC	–	–	–	–	–	–	$\underline{1.63}\times10^{-5}$

5.3. Evaluation in terms of moderate scoliosis and severe scoliosis

To better reflect the accuracy of the eight methods, we further evaluate the performance based on moderate scoliosis and severe scoliosis (see Fig. 20 in Appendix 1). For moderate scoliosis, Team PAT got the highest value of 0.76, and Team iFLYTEK got the second-highest value of 0.73. Team PAT added an additional rotation angle parameter θ to the bounding box detection, which may make their method better for relatively large curvature of the spine. Team iFLYTEK performs vertebra prediction and keypoint detection in two lines in parallel, which may also enable a better prediction of the severely curved spine. The performances based on severe scoliosis is shown in Fig. 21 (see Appendix 1). Basically, all teams except Team UC have achieved good results.

In general, the algorithm in an end-to-end way is better in terms of overall performance. However, for mild spine bending angles, Team ErasmusMC's method may bring more accurate calculations. For the spine with large deformation, it is more difficult to find the landmarks accurately. The prediction needs to consider the influence of the degree of deformation. Furthermore, we evaluated the best and worst images predicted by each team on 5 metrics (SMAPE, CMAE, ED, MD, and CD) and the top 2 images are shown in Fig. 16. We also visualized the best and worst cases in Fig. 17 for two teams. It can be seen that for images with blurry spine boundaries and large deformations, it is a difficulty in the current algorithm. For the spine that is basically not deformed, the effect of the algorithm is better.

5.4. Limitation and future work

In terms of challenge evaluation, we only selected SMAPE as the challenge metric, and may not be able to evaluate the performance of methods very well. Therefore, we will consider more metrics and provide in the future challenge, such as CMAE, etc. Moreover,

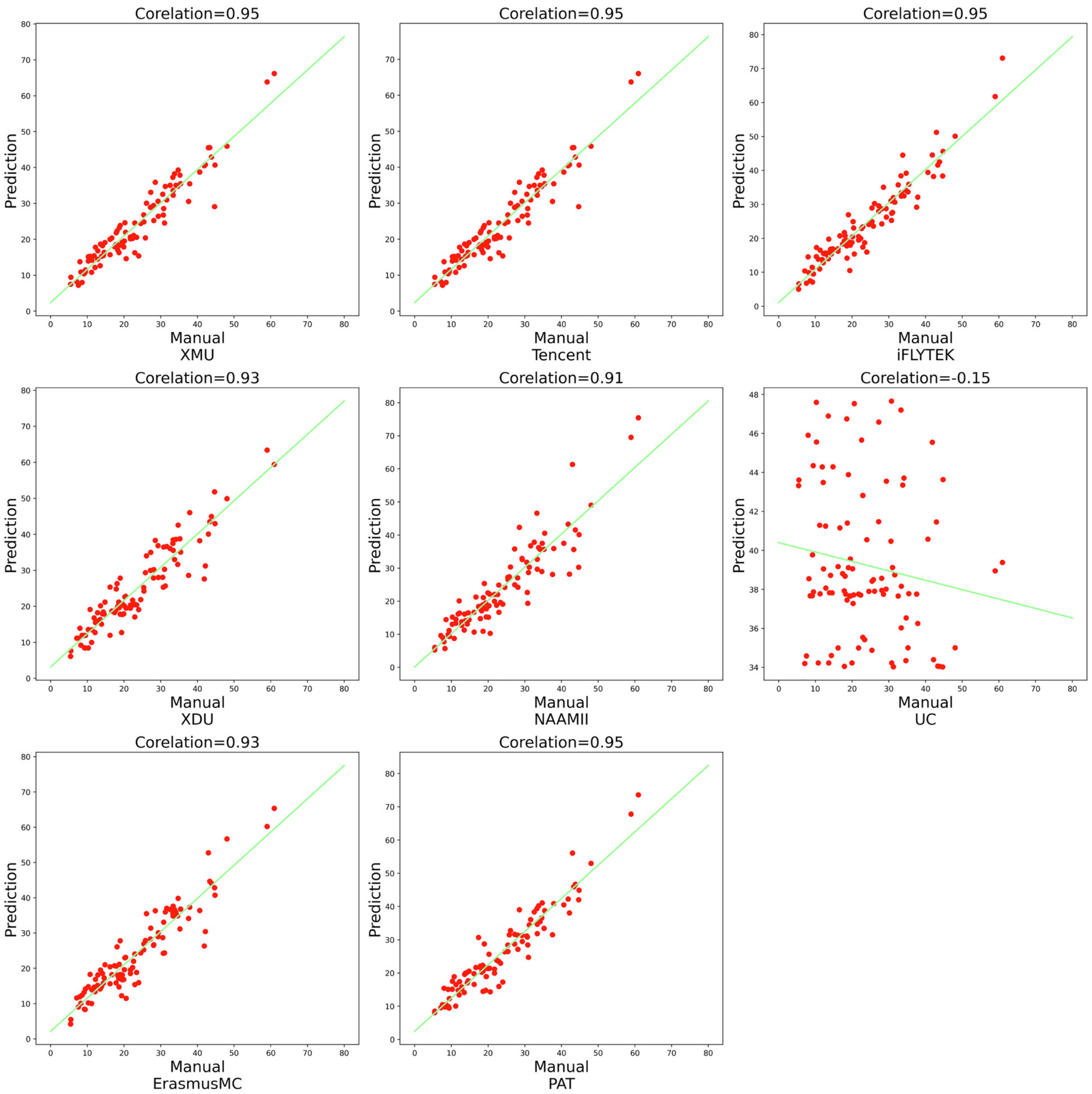

Fig. 15. The closeness between the ground truth and the angle predicted by the proposed eight methods in the max angle value of all scoliosis ranges. It can be seen that most methods show a high correlation.

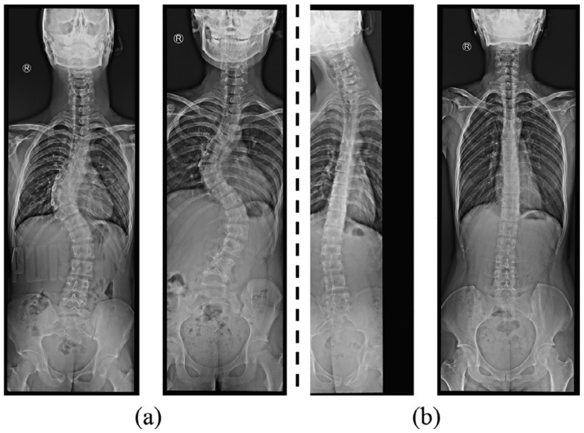

Fig. 16. Best and worst cases in the testing dataset. (a) the worst 2 images, and (b) the best 2 images.

the testing set only provides the Cobb angles as the ground truth for the challenge. Given that each team considers the shape of the spine into their methods, we think that assessing the shape of the spine is also important. An assessment of the landmark or the centerline of the spine will be added.

In terms of data, the testing set and training set we provide are somewhat different in distribution, which increases the difficulty of the challenge. The teams need to find a way to adapt to the difference. The images we provided are the JPG format, resulting in a somewhat lower resolution. In the future, we hope to provide higher resolution images. Besides, the number of training and testing images are small. We are working hard to get more images.

In the paper, we only reviewed 8 teams. Some works are not included due to time limits. We look forward to more scholars joining the research of automatic spine curve estimation in the future.

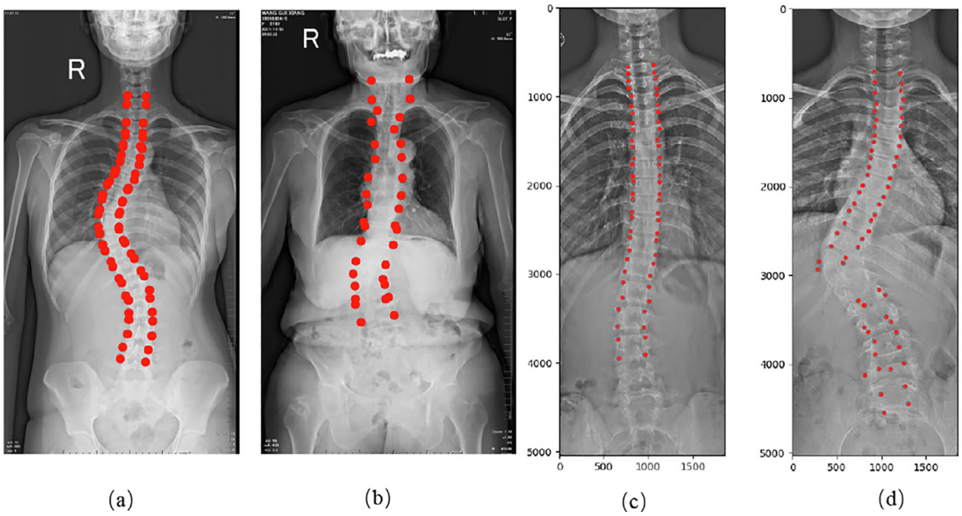

Fig. 17. Visualization of the worst and best case scenarios for two teams with respect to the algorithmic performance. (a) and (b) are the best and worst cases for the Team NAAMII. (c) and (d) are the best and worst cases for the Team XDU.

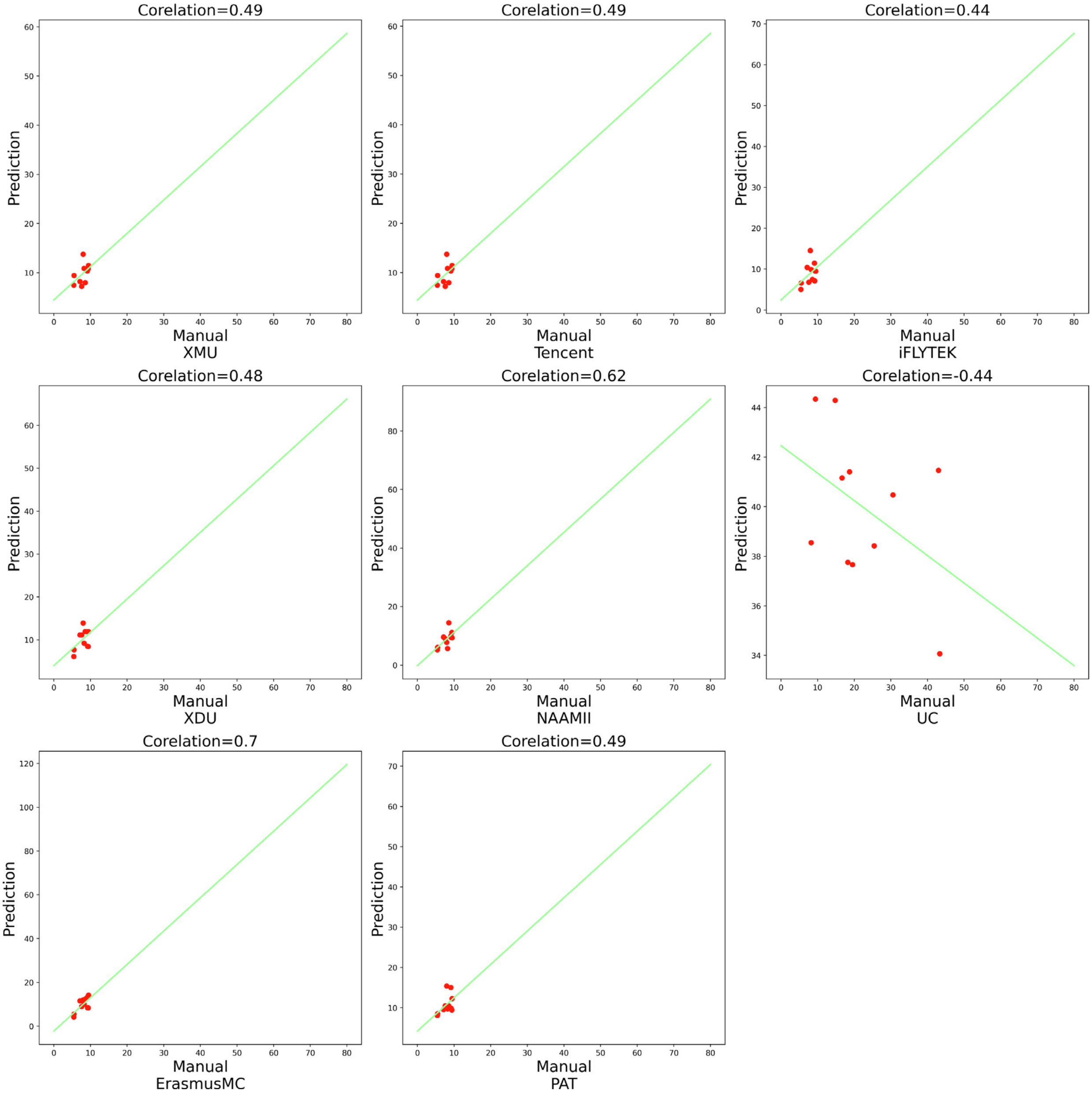

Fig. 18. The closeness between the ground truth and the angle predicted by the proposed eight methods in the max angle value of not scoliosis. It can be seen that most methods show a high correlation.

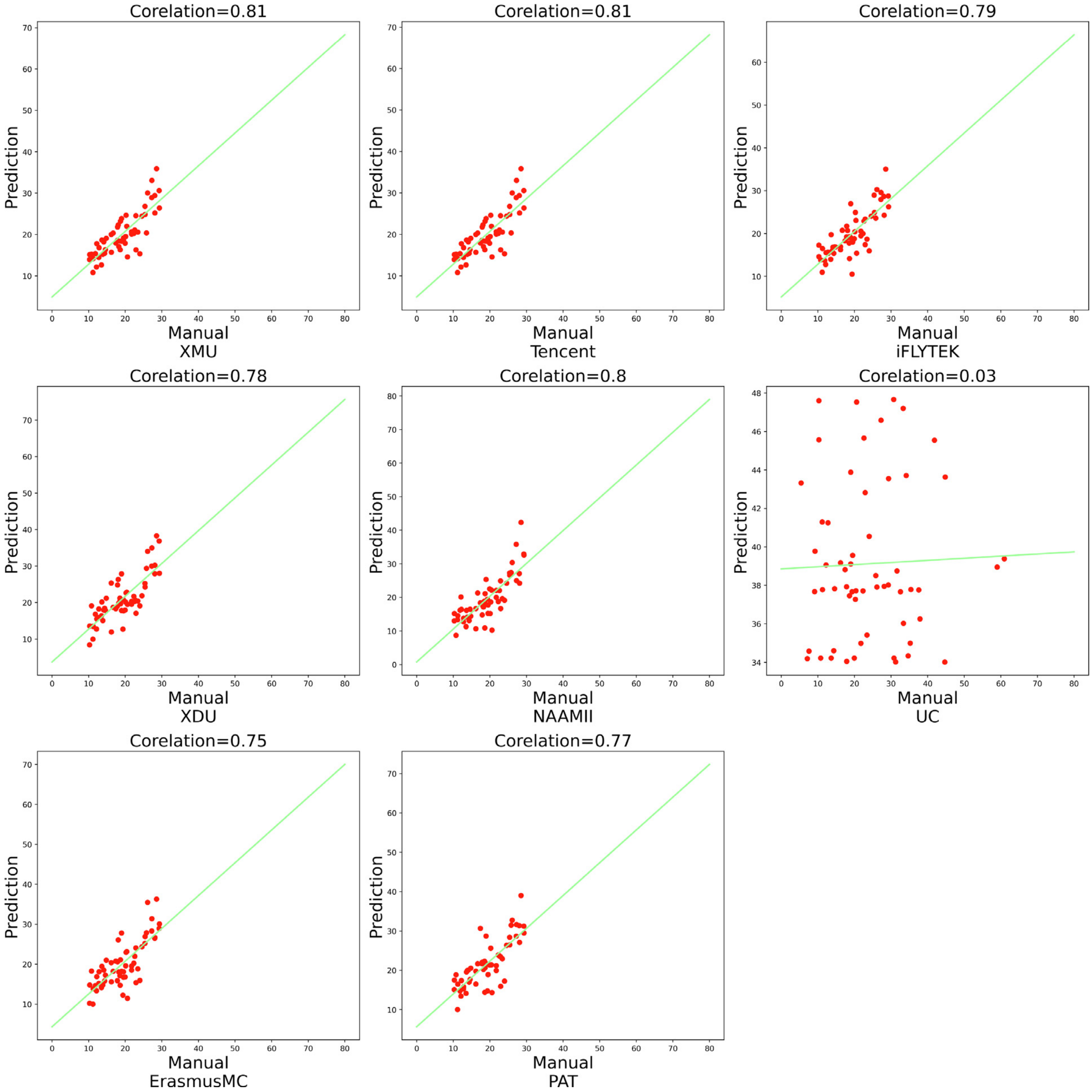

Fig. 19. The closeness between the ground truth and the angle predicted by the proposed eight methods in the max angle value of mild scoliosis. It can be seen that most methods show a high correlation.

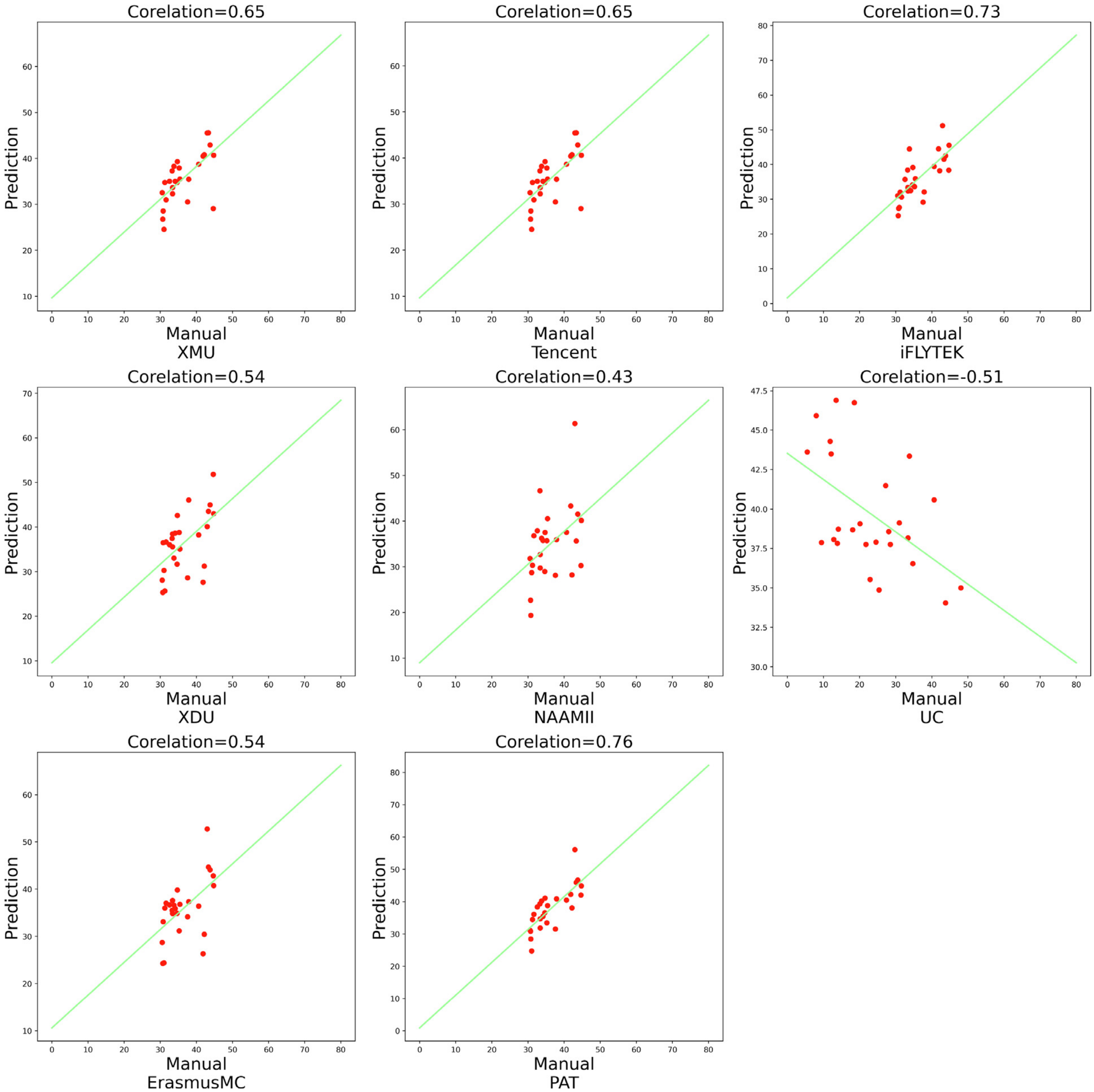

Fig. 20. The closeness between the ground truth and the angle predicted by the proposed eight methods in the max angle value of moderate scoliosis. It can be seen that most methods show a high correlation.

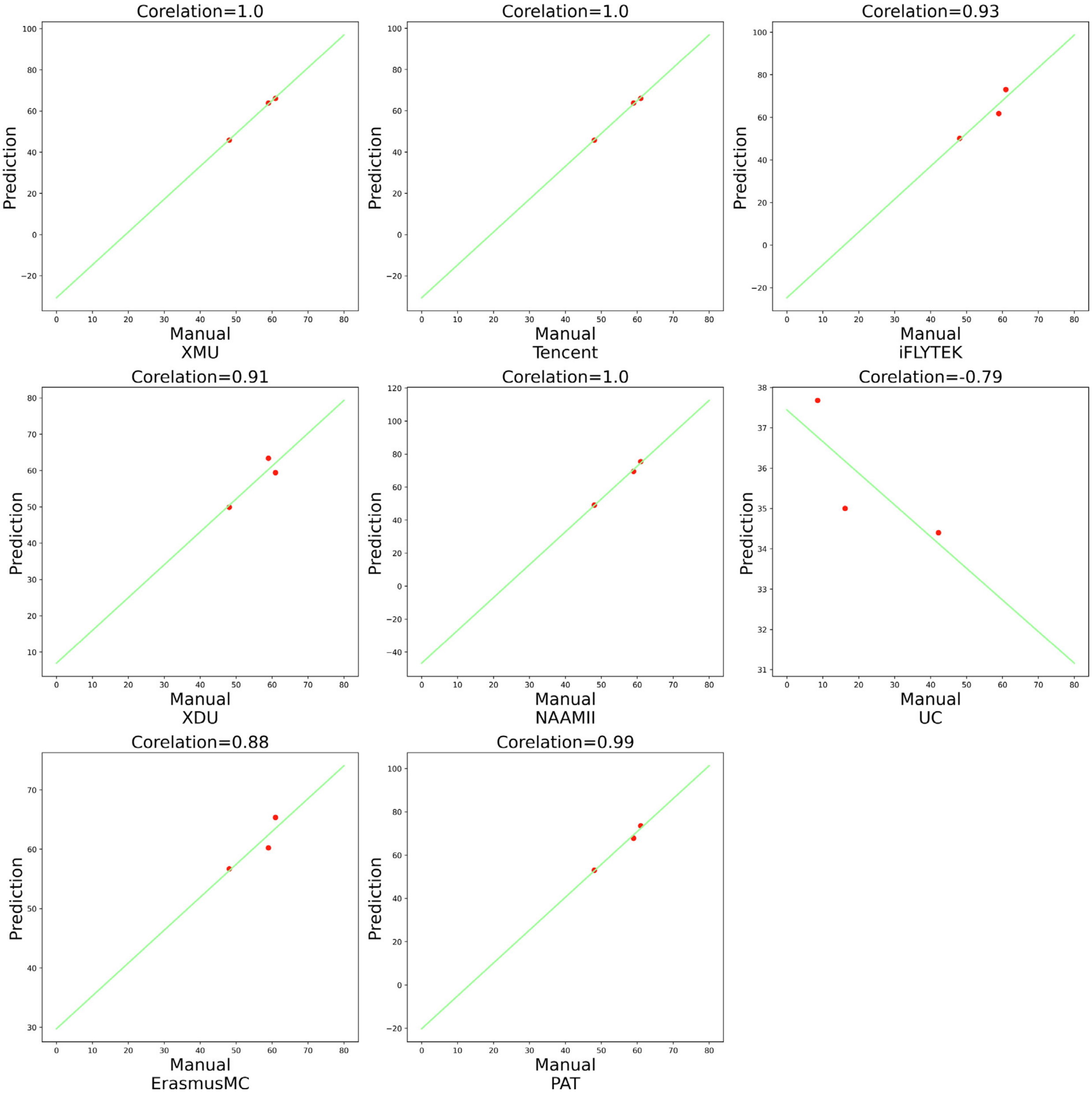

Fig. 21. The correlation coefficients between the ground truth and the angle predicted by the proposed eight methods in the max angle value of severe scoliosis. It can be seen that most methods show a high correlation.

6. Conclusion

In this paper, we have elaborated on the details of eight methods, including intuition, workflow, and implementation. All eight methods were presented to the AASCE-2019 challenge. We further analyze the performance of these methods under different metrics, most of which can achieve excellent results. A detailed discussion is provided with limitations and possible future directions. This paper will provide possible solutions and insightful discussion for researchers who want to continue working on the Cobb angle estimation from x-ray images. Organizers choose not to disclose the corresponding ground truth of test data. The challenge will reopen for new submissions and can thus be used as a reference for algorithm performance on spinal curvature estimation.

Declaration of Competing Interest

None.

Appendix A. Appendix 1: Evaluation results in terms of different kinds of scoliosis

References

Aubert, B., Vidal, P., Parent, S., Cresson, T., Vazquez, C., De Guise, J., 2017. Convolutional neural network and in-painting techniques for the automatic assessment of scoliotic spine surgery from biplanar radiographs. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 691–699.

Barton, C.B., Weinstein, S.L., 2018. Adolescent idiopathic scoliosis: natural history. In: Pathogenesis of Idiopathic Scoliosis. Springer, pp. 27–50. doi:10.1007/978-4-431-56541-3_2.

Bloch, K.E., Palange, P., Simonds, A., 2012. ERS Handbook: Self-Assessment in Respiratory Medicine. European Respiratory Society.

Chen, B., Xu, Q., Wang, L., Leung, S., Chung, J., Li, S., 2019. An automated and accurate spine curve analysis system. IEEE Access 7, 124596–124605.

Cheng, J.C., Castelein, R.M., Chu, W.C., Danielsson, A.J., Dobbs, M.B., Grivas, T.B., Gurnett, C.A., Luk, K.D., Moreau, A., Newton, P.O., et al., 2015. Adolescent idiopathic scoliosis. Nat. Rev. Dis. Primers 1 (1), 1–21.

Cobb, J., 1948. Outline for the study of scoliosis. Instr Course Lect AAOS 5, 261–275.

Ganin, Y., Lempitsky, V., 2015. Unsupervised domain adaptation by backpropagation. In: Proceedings of the 32nd International Conference on Machine Learning. PMLR, pp. 1180–1189.

He, K., Gkioxari, G., Dollár, P., Girshick, R., 2017. Mask R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2961–2969.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778.

Horng, M.-H., Kuok, C.-P., Fu, M.-J., Lin, C.-J., Sun, Y.-N., 2019. Cobb angle measurement of spine from x-ray images using convolutional neural network. Comput. Math. Methods Med. 2019.

Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q., 2017. Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700–4708.

Illés, T.S., Lavaste, F., Dubousset, J.F., 2019. The third dimension of scoliosis: the forgotten axial plane. Orthopaed. Traumatol. 105 (2), 351–359.

Ioffe, S., Szegedy, C., 2015. Batch normalization: accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning. PMLR, pp. 448–456.

Irvin, J., Rajpurkar, P., Ko, M., Yu, Y., Ciurea-Ilcus, S., Chute, C., Marklund, H., Haghgoo, B., Ball, R., Shpanskaya, K., et al., 2019. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, pp. 590–597.

Kenneth, S.S., 2017. Anatomy & Physiology: the Unity of Form and Function, 5. McGraw-Hill, New York, NY, USA:.

Lau, K., 2013. The Complete Scoliosis Surgery Handbook for Patients: an In-depth and Unbiased Look Into what to Expect Before and During Scoliosis Surgery. Health In Your Hands.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017. Focal loss for dense object detection. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2980–2988.

O'Brien, M., Kuklo, T., Blanke, K., Lenke, L., 2008. Radiographic Measurement Manual. Spinal Deformity Study Group. Medtronic Sofamor Danek USA Inc., Memphis, TN.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in neural information processing systems, pp. 91–99.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 234–241.

Sun, H., Zhen, X., Bailey, C., Rasoulinejad, P., Yin, Y., Li, S., 2017. Direct estimation of spinal cobb angles by structured multi-output regression. In: International Conference on Information Processing in Medical Imaging. Springer, pp. 529–540.

Sun, K., Xiao, B., Liu, D., Wang, J., 2019. Deep high-resolution representation learning for human pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5693–5703.

Tan, M., Le, Q., 2019. Efficientnet: rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning. PMLR, pp. 6105–6114.

Wang, L., Xu, Q., Leung, S., Chung, J., Chen, B., Li, S., 2019. Accurate automated cobb angles estimation using multi-view extrapolation net. Med. Image Anal. 58, 101542. doi:10.1016/j.media.2019.101542.

Xiao, B., Wu, H., Wei, Y., 2018. Simple baselines for human pose estimation and tracking. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 466–481.

Zeiler, M. D., 2012. Adadelta: an adaptive learning rate method. arXiv:1212.5701.

Zhang, K., Xu, N., Yang, G., Wu, J., Fu, X., 2019. An automated cobb angle estimation method using convolutional neural network with area limitation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 775–783.

Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J., 2017. Pyramid scene parsing network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2881–2890.