



# Next-generation prognosis framework for pediatric spinal deformities using bio-informed deep learning networks

Mahsa Tajdari<sup>1</sup> · Farzam Tajdari<sup>2</sup> · Pouyan Shirzadian<sup>3</sup> · Aishwarya Pawar<sup>4</sup> · Mirwais Wardak<sup>5</sup> · Sourav Saha<sup>6</sup> · Chanwook Park<sup>6</sup> · Toon Huysmans<sup>2</sup> · Yu Song<sup>2</sup> · Yongjie Jessica Zhang<sup>7</sup> · John F. Sarwark<sup>8</sup> · Wing Kam Liu<sup>1</sup>

Received: 4 February 2022 / Accepted: 2 September 2022 / Published online: 11 October 2022  
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2022

## Abstract

Predicting pediatric spinal deformity (PSD) from X-ray images collected on the patient's initial visit is a challenging task. This work builds on our previous method and provides a novel bio-informed framework based on a mechanistic machine learning technique with dynamic patient-specific parameters to predict PSD. We provide a geometry-based bone growth model that can be utilized in a range of applications to enhance the bio-informed mechanistic machine learning framework. The proposed technique is utilized to examine and predict spine curvature in PSD cases such as adolescent idiopathic scoliosis. The best fit of a segmented 3D volumetric geometry of the human spine acquired from 2D X-ray images is employed. Using an active contour model based on gradient vector flow snakes, the anteroposterior and lateral views of the X-ray images are segmented to derive the 2D contours surrounding each vertebra. Using minimal user input, the snake parameters are calibrated and automatically computed over the dataset, resulting in fast image segmentation and data collection. The 2D segmented outlines of each vertebra are transformed into a 3D image segmentation result. The Iterative Closest Point mesh registration technique is then used to establish a mesh morphing approach and creates a 3D atlas spine model. Using the comprehensive 3D volumetric model, one can automatically extract spinal geometry data as inputs to the mechanistic machine learning network. Moreover, the proposed bio-informed deep learning network with the modified bone growth model achieves competitive or even superior performance against other state-of-the-art learning-based methods.

**Keywords** Mechanistic bio-informed machine learning · X-ray images · Dynamic model · Pediatric spinal deformity · Adolescent idiopathic scoliosis · Mesh registration · 3D reconstruction

## 1 Introduction

Deformities in the spine may cause pain, numbness, tingling sensation, loss of function, and even pulmonary and cardiac difficulties. If the spine loses its usual, graceful S-shape (when seen from the side), or if the spine loses its

---

Mahsa Tajdari and Farzam Tajdari contributed equally to this work.

✉ Yongjie Jessica Zhang  
jessicaz@andrew.cmu.edu

✉ Wing Kam Liu  
w-liu@northwestern.edu

<sup>1</sup> Department of Mechanical Engineering, Northwestern University, Evanston 60208, IL, USA

<sup>2</sup> Faculty of Industrial Design Engineering, Technical University of Delft, Mekelweg 5, Delft 2628 CD, Netherlands

<sup>3</sup> Department of Computer Science, Virginia Tech, Blacksburg, VA 24061, USA

<sup>4</sup> School of Mechanical Engineering, Purdue University, West Lafayette, IN 47907, USA

<sup>5</sup> Stanford Cardiovascular Institute and Department of Radiology, Stanford University, Stanford, CA 94305, USA

<sup>6</sup> Theoretical and Applied Mechanics, Northwestern University, Evanston, IL 60208, USA

<sup>7</sup> Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, USA

<sup>8</sup> Department of Orthopaedic Surgery, Northwestern University Feinberg School of Medicine, Chicago, IL 60611, USA

straightness, deformities may arise (viewed from front to rear). The term Pediatric Spinal Deformity (PSD) refers to such malformations of the spine in children, namely, scoliosis, kyphosis, and spondylolisthesis [1]. Scoliosis and kyphosis are conditions characterized by aberrant curvature of the spine, while spondylolisthesis is characterized by the displacement of vertebrae. PSD is caused by a variety of factors, the most important of which is bone growth, which controls the curvature and advancement of the deformity. There is still more to learn about the underlying etiology of PSD. Affecting roughly 7 million people in the United States, scoliosis is one of the most prevalent spinal abnormalities. Adolescent Idiopathic Scoliosis (AIS) is used to describe the condition in children and adolescents. AIS accounts for around 80% of all pediatric scoliosis cases, while affecting approximately 3% of adolescents under the age of 16 in the USA [2].

The treatment of scoliosis is mainly reliant on the shape and extent of spinal curvature, and specific treatment choices are usually determined by the surgeon's expertise. As a result, the development of a clinically validated, patient-specific model of the spine to assist surgeons in the prognosis of early-stage PSD would guide optimum surgical and non-surgical treatment options. For both screening and monitoring in present clinical practice, the lack of an appropriate safe, inexpensive, and accurate measuring technology is a major bottleneck. To track the evolution of deformities throughout adolescent growth, frequent imaging is essential [3] and the extracted features from medical images such as Cobb angle determine the severity of spinal deformity. The Cobb angle is the most often utilized measurement for determining the severity of spinal deformities.

It is common practice in spine surgery to use computer-aided procedures, such as determining the best path for the insertion of pedicle screws [4–6], as well as improved surgical navigation and a more comprehensive pre-operative surgical plan [7, 8]. One of the greatest barriers to incorporating these technologies into clinical practice is the time and effort necessary to generate patient-specific functional models from medical imaging. It entails a number of manual procedures and is time intensive, even for seasoned specialists [9, 10]. For example, image segmentation is both time-consuming and user-dependent [11, 12] because it requires to locate and segment vertebrae ahead of time [14]. Generating 3D detailed geometry of the spine from a 2D set of X-ray images is also a challenging task. Various manual processes are required to segment the obtained volumetric mesh in order to detect hard and soft tissue once the 3D model has been formed.

Machine learning (ML) approaches need a large amount of data to be trained and provide reliable results [14, 15]. The absence of reliable medical data for a given individual over time is one of the challenges in implementing ML for

prognosis spinal deformity. Furthermore, these models are referred to as “data-hungry” approaches since they cannot forecast outside of the range of the training data [16]. Recent publications show that by including physics into the system, one may enhance the model's predictability range [17]. However, for spinal curve progression including bone formation, patient-specific governing physical equations with time-varying and geometric-based coefficients are unavailable.

Previously published research from our group has used X-ray imaging data to develop a prognostic framework for AIS [18]. This framework predicted spine morphology by combining clinical data acquired from X-ray images with mechanistic features such as stress distribution on the growing surface of the vertebrae extracted from a spine surrogate model along with the bone growth model. Although this unique framework used mechanistic data science for forecasting spine deformity, the previous method had a number of limitations that impede deploying this framework in real-world scenarios. One restriction is the manual parameter adjustment for image segmentation of each vertebra. The presented 3D reconstruction and geometry generation technique requires considerable labor since each tissue is generated separately and all tissues are assembled together. Moreover, the bone growth model has constant parameters throughout all patients and time steps. However, these assumptions are unrealistic since bone formation differs across ages and vertebrae.

The proposed framework (Fig. 1) in this paper builds on our previous work [18], with each step in the framework improved to meet real-world needs. Both the limitations of the previous framework and the contributions of the current framework are summarized in Table 1.

The organization of the paper is as follows: After discussing related works in Sect. 2, we describe the framework to create the patient-specific 3D geometry from X-ray images in Sect. 3. In Sect. 4, we present our novel bio-informed mechanistic machine learning model for prognosis of pediatric spinal deformity. Lastly, we end with some conclusions and discuss future directions of this work.

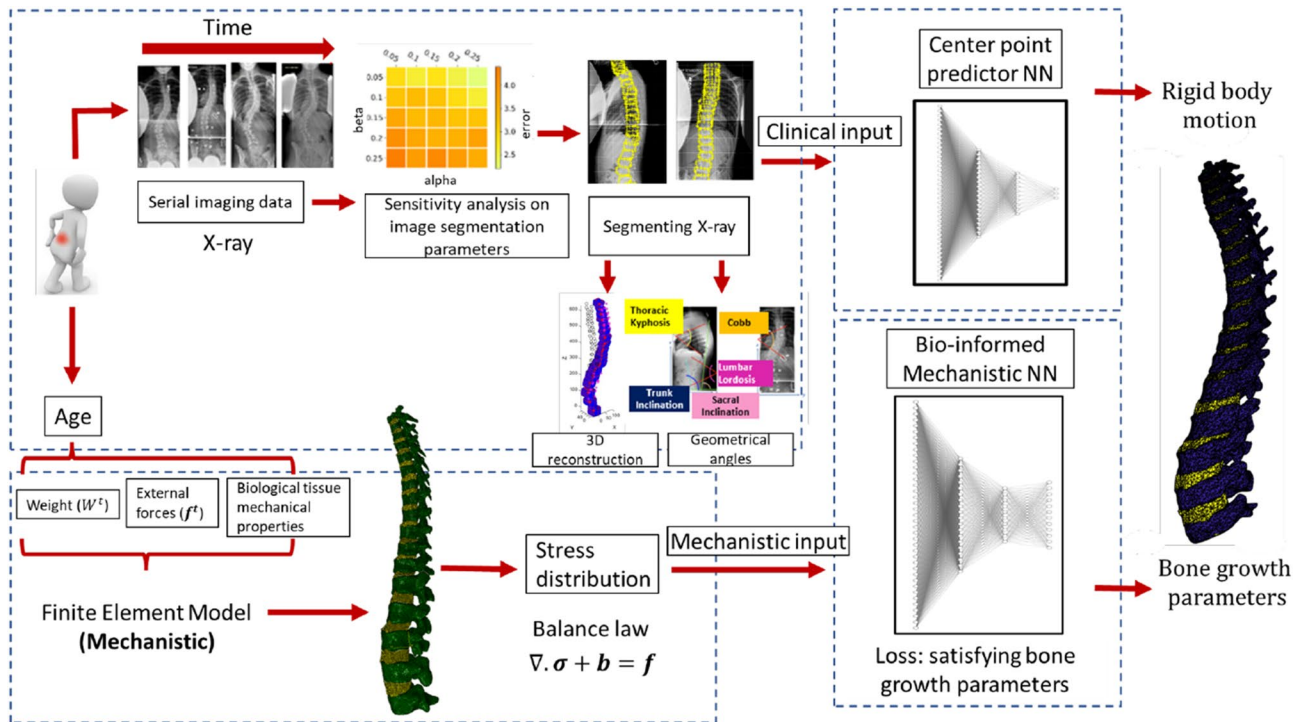
## 2 Related works

### 2.1 Vertebrae image segmentation

Precisely segmenting the vertebrae is critical for subsequent analysis in an injury detection system. Statistical shape model (SSM)-based techniques have dominated previous work in vertebral segmentation [19, 20]. Based on a training set, these approaches capture statistical information on the shape and/or appearance of the vertebra. The mean shape is then manually or semi-automatically set close to the

**Table 1** Contributions of proposed framework with respect to previous framework [18]

Steps of framework	Limitations of previous framework	Contributions of current framework
1 2D image segmentation	Manual parameter adjustment	Calibrated parameters and semi-manual adjustment to expedite the process
2 3D reconstruction from bi-planar data and volumetric mesh generation	Labor intensive	Fast and robust
3 Mechanistic deep learning method using bone growth model	Constant parameters which are independent of time and position	Patient-specific, position-sensitive and dynamic bio-informed mechanistic deep learning parameters



**Fig. 1** The overall workflow of the proposed framework. The geometric data of vertebrae is retrieved from 2D X-ray time-series images using an image segmentation algorithm whose parameters are calibrated through sensitivity analysis. The 2D data is then translated

real vertebra, and a search process is used to converge the shape on the true vertebral boundaries. Latest evidence has used random forest-based machine learning (ML) models to achieve shape convergence [20–23]. These approaches, however, are only efficient and accurate for the restricted data reported in the literature and cannot be applied to patient-specific datasets.

### 2.2 3D model development and computational simulation

Compared to MRI and computational tomography (CT) scanners, X-ray images are more commonly employed due to their

accessibility, lower cost, shorter scanning time, and lower ionizing radiation levels particularly for adolescent patients. Statistical Shape Models (SSM) [24–27] or Statistical Shape and Intensity Models (SSIM) were used to reconstruct bones from X-ray images. To prevent local maxima while optimizing the deformable model parameters, it is important to select a reasonable starting point [28, 29]. Recent years have seen the use of deep learning to recognize landmarks and triangulate them [28]. However, performing 3D reconstruction from two or more 2D photographs using a deep learning technique remains a tough problem due to the complexity of describing a dimensional expansion in multi-view circumstances. The EOS imaging system (formerly, Biospace Med, Paris, France), the

to 3D data where patient-specific features are extracted. The imaging data (clinical) and the mechanistic aspects such as the dynamic patient-specific bone growth model are passed through the bio-informed mechanistic neural network to predict spinal deformity

DIERS formetric scanner, and ultrasonography are examples of recent breakthroughs in diagnostic imaging for AIS [3]. The EOS imaging system is made up of two orthogonal pairs of X-ray tubes and detector units that allow for the simultaneous capture of anteroposterior (AP) and lateral (LAT) X-ray pictures while standing. EOS imaging can quickly scan the spine in 8–15 s, depending on the patient's height. From the EOS anteroposterior and lateral pictures, the software system sterEOS (EOS Imaging, Paris, France) can generate a highly accurate 3D model of the spine [30–32]. However, this technology is unable to divide distinct tissues in order to account for the many biological organs. Due to its expensive cost, it is not widely available in many medical centers.

To gain a better knowledge of load distribution and other mechanical features, the 3D generated patient-specific geometry may be utilized to construct a finite element model. There is no comprehensive automated workflow for anatomically correct FE simulations of the spine based on 2D X-ray data. A lot of work has been done on parametric FE models or a mix of statistical and FE models [33, 34]. However, those models either ignore essential patient-specific features or require a lot of manual labor, which necessitates a certain level of operator experience. Although efforts to automate the construction of FE models of the healthy spine have been performed [10, 35], the technique has never been integrated with deep learning-based segmentation algorithms or applied to diseased situations.

### 2.3 Implementing ML for studying spinal deformity

The application of ML in medical research has skyrocketed in recent decades. When it comes to applying ML for medical image analysis, there has always been the challenge on how to accurately integrate ML for disease diagnosis, prognosis, and therapy. A framework with such characteristics should always be able to capture the biological governing equation in order to offer extra information to the lacking training data. Recent studies have attempted to use AI to predict spinal deformities [36–39]. These frameworks, however, cannot be applied to other disciplines. Recently, research has shown that by incorporating the system's underlying physical equations, the framework may forecast data outside of the projected range [18, 40]. Nevertheless, there are certain processes in between that need manual parameter adjustment and, as a result, cannot be implemented for real-time prognosis framework.

## 3 Patient-specific image segmentation and data generation

### 3.1 Image segmentation and parameter fitting

Image segmentation of clinical X-ray images is carried out to extract features for the prognosis framework [41]. The

corner points of each vertebra are identified and used as reference points to monitor variations in spine shape and bone formation over time. We implement semi-automated image segmentation using active contour, also known as, the snakes method [42]. A rectangular contour consisting of four corner points is initialized manually around each vertebra and evolved to capture the shape. We define the four corner points of each vertebra as landmarks. These are the key points which help in describing the spine geometry. The segmentation is carried out in 2D for both AP and LAT images. The evolution of the active contour is carried out iteratively through minimization of image energy, allowing it to converge at the edges of features. Snakes are considered energy-minimizing splines which are defined in a parametric form. The total energy functional proposed in [42] considers both image and external constraint energy terms (described in detail in Appendix A).

Active contour model is fast and is able to accurately segment each vertebra in the X-ray images. However, there are a few drawbacks. The accuracy of segmentation depends on how accurately the active contour is initialized. The energy functional which is minimized consists of many weighting parameters which are set empirically at the beginning of the segmentation. The optimized parameters of the snakes method which provides the contour around each vertebra are generally obtained through a trial and error process, which is time-consuming. We propose an improved technique to obtain optimum parameters for the accurate segmentation of clinical X-ray images. The parameters of the Snake algorithm are categorized into two groups: geometric parameters and fitting parameters.

*Geometric parameters* For each vertebra, we initialize the active contour as a rectangle and define geometric parameters such as the width ( $w_n$ ), height ( $h_n$ ) and rotation angle ( $\theta_n$ ) from the horizontal axis.  $n$  is the vertebra level number where  $n = 1 \sim 12$  corresponds to the thoracic vertebra (T1, T2, ..., T12) and  $n = 13 \sim 17$  represent lumbar vertebra (L1, L2, ..., L5). From the T1 to the L5 vertebra, we assume a linear relationship between each vertebra size and vertebra level, with L5 vertebra having the largest size. The length and width of T1 and L5 can be used to determine the corresponding length and width of other vertebrae. The user chooses three reference corner points (top right, bottom right and bottom left) for T1 ( $w_1$  and  $h_1$ ) and L5 ( $w_m$  and  $h_m$ ) vertebrae as shown in Fig. 2a, and the parameters for the remaining vertebrae are computed using

$$\begin{aligned} h_n &= h_1 + \left( \frac{h_m - h_1}{N - 1} \right) (n - 1), \\ w_n &= w_1 + \left( \frac{w_m - w_1}{N - 1} \right) (n - 1). \end{aligned} \quad (1)$$

For each vertebra, the user manually selects the center point. The rotation angle can be obtained by

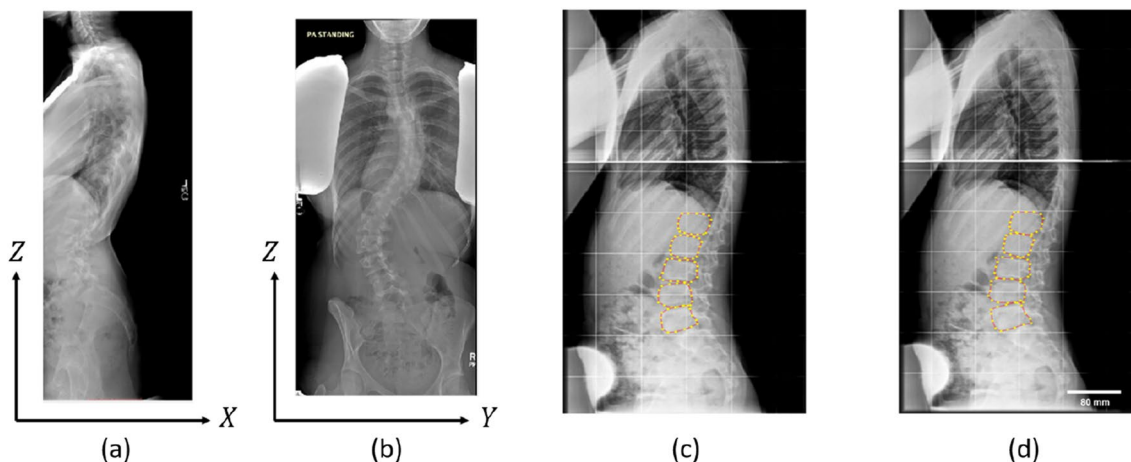
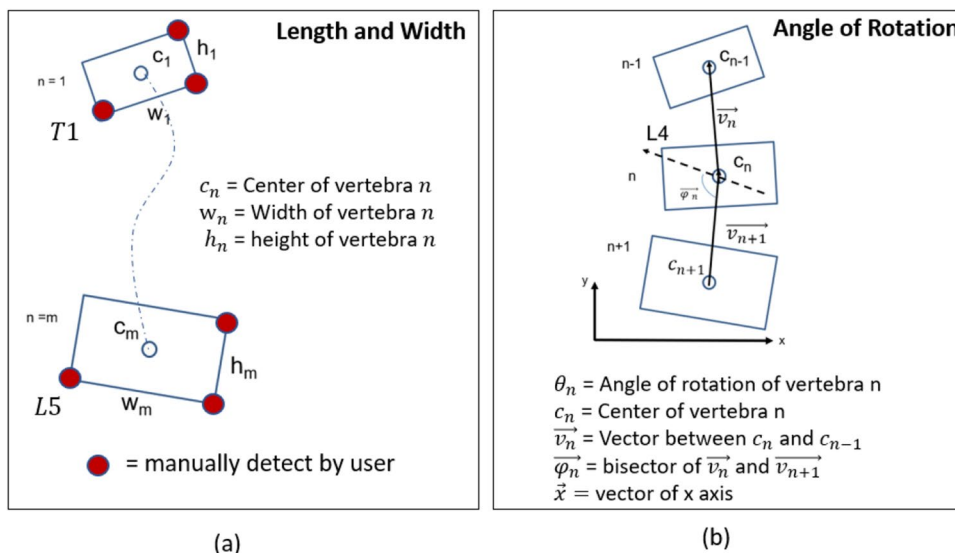
$$\theta_n = \cos^{-1} \left( \frac{\vec{\phi}_n \cdot \vec{x}}{|\vec{\phi}_n| \cdot |\vec{x}|} \right) \tag{2}$$

where  $\theta_n$  is the rotation angle of the vertebra  $n$ ,  $\vec{x}$  is the vector of horizontal axis and  $\vec{\phi}_n$  is the bisector of the vector that connects the two adjacent vertebrae as shown in Fig. 2b. It is worth noting that the framework may be used for any number of vertebrae in any regions of interest, including cervical, thoracic, lumbar, or a combination of all three. By estimating geometric parameters for the lumbar spine, Fig. 3 compares the ground truth and the modified snake method. The findings are in excellent accord with the ground truth,

as shown. The ground truth segmentation result is evaluated using the snakes method in which the parameters are manually adjusted and the initialization of the contour is done manually for each vertebra.

*Fitting parameters* The weighting coefficients in the active contour model are set in order to move the contour around each vertebra.  $\alpha$  and  $\beta$  are the weights associated with the first-order and second-order regularizing terms of the internal spline energy (Eq. (2) in [42]) that control the tension and rigidity of the snake, respectively.  $\gamma$  is the step size associated with the iterative update of the active contour and  $\kappa$  is the weighting coefficients associated with the derivatives of the external force terms (Eq. (17) in [42]).  $w_e$ ,  $w_l$  and  $w_t$  are the weighting coefficients associated with the image, edge and terminal energy functionals, respectively

**Fig. 2** **a** Approximating length and width of vertebrae by picking reference corner points of T1 and L5; and **b** the approximation of angle of rotation for each vertebra



**Fig. 3** Segmentation of the X-ray images. **a** Lateral (LAT) and **b** anteroposterior (AP) view of the image obtained from X-ray data. Comparison between two segmented lumbar spine: **c** ground truth

(the output of the snakes algorithm in which the parameters are manually adjusted) and **d** by estimating geometric parameters (angle of rotations, length and width of each vertebra)

[42]. More details on the fitting parameters are provided in Appendix A. By identifying the optimal sets of these parameters, one may precisely determine the curvature of the spine.

A sensitivity analysis [43, 44] is performed on each pair of fitting parameters, namely  $\alpha$ - $\beta$ ,  $\gamma$ - $\kappa$ ,  $w_l$ - $w_e$  and  $w_l$ -iterations, as illustrated in Fig. 4a–d, to provide a better estimate on the optimized parameters. Here, for a particular pair of fitting parameters, the segmentation result is compared with the ground truth result and the average value of the normalized mean squared error is calculated for all the images in the dataset. The normalized mean squared error is given as

$$\frac{1}{|N_s| \cdot |N_v| \cdot |N_\ell|} \sum_{s=1}^{N_s} \sum_{v=1}^{N_v} \sum_{\ell=1}^{N_\ell} \left( (x_{v,\ell}^s - \bar{x}_{v,\ell}^s)^2 + (y_{v,\ell}^s - \bar{y}_{v,\ell}^s)^2 \right) \tag{3}$$

where  $N_s$ ,  $N_v$  and  $N_l$  are the number of images, number of vertebra and number of landmarks associated with each vertebrae.  $x_{v,l}^s$  and  $y_{v,l}^s$  are the positions of landmarks obtained by the current approach, whereas  $\bar{x}_{v,l}^s$  and  $\bar{y}_{v,l}^s$  are the positions of the landmarks obtained from ground truth result.

To obtain the optimized parameters throughout the dataset, each pairwise parameter with the least normalized mean squared error is the initial value in the Simplex optimization. The final set of optimized parameters are used to segment the images, as shown in Fig. 4e. We observe that the method

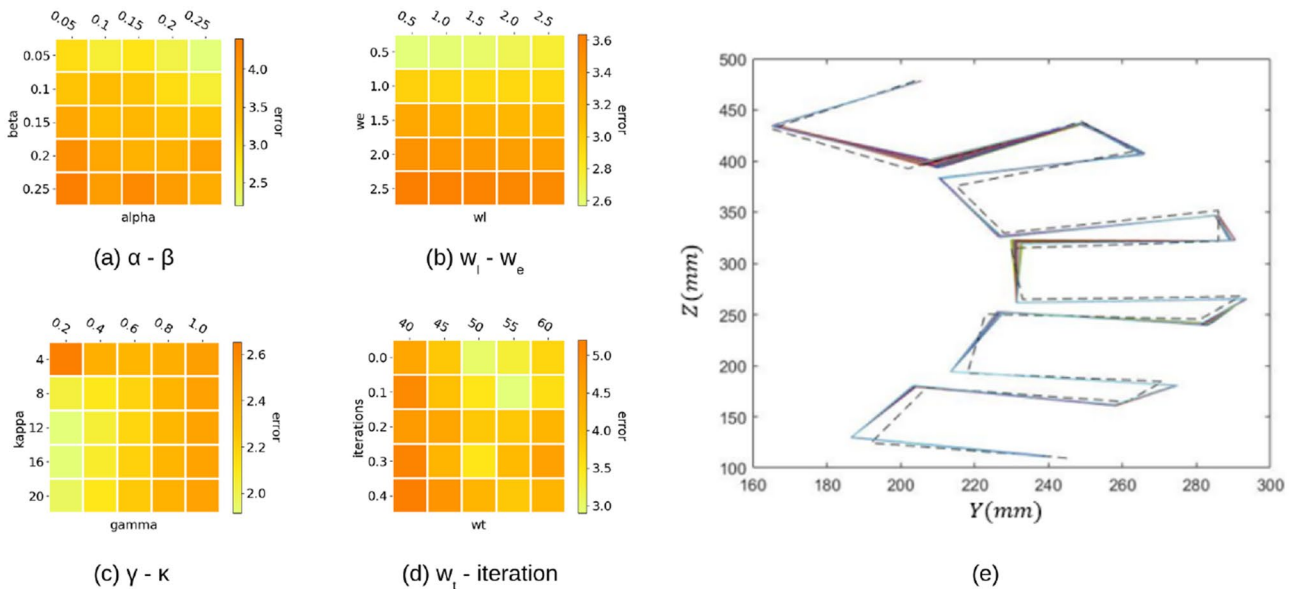
is not very sensitive to the parameter values, which is desirable for practical applications, since there is no need for fine tuning of parameters in order to achieve satisfactory results.

Figure 5 shows the segmented spine in AP view, taking into account both the geometric and fitting parameters. First, the reference points of the T1 and L5 vertebrae will be identified and then the program will identify the contour around each vertebra.

### 3.2 3D shape reconstruction from bi-planar 2D data

Serial X-ray images from two orthogonal perspectives known as anteroposterior (AP) and lateral (LAT) are used in this study to gather patient-specific data (Fig. 3). Patient-specific characteristics are retrieved from X-ray images to assess spinal deformity prognosis. After performing segmentation and obtaining the contours from AP and LAT images, we perform 3D reconstruction from the 2D segmented contours in order to get the volumetric spine geometry. The different coordinate systems associated with the AP and LAT images present a challenge that needs to be addressed. Furthermore, it is challenging to keep track of the various coordinate systems of these images that have been longitudinally taken over time.

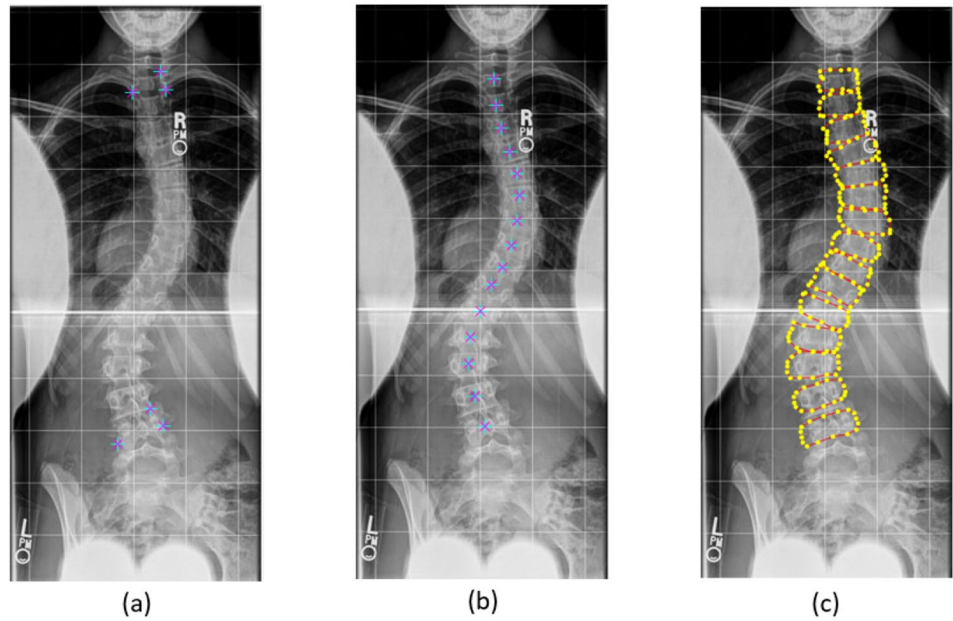
The initial step should be to calibrate the camera location in order to make all data consistent and comparable in scale. Here are the assumptions for the camera calibration step:



**Fig. 4 a–d** Sensitivity analysis on the *fitting parameters* of the snakes algorithm to be provided as initial values for multi-variable optimization. **e** Optimized multi-variables of the snakes method for the lumbar spine. The dashed line denotes the ground truth manually picked

four corner points of the vertebrae, and solid lines are the optimized configuration. The normalized mean square error is 0.0187 corresponding to  $\alpha = 0.25$ ,  $\beta = 0.25$ ,  $\gamma = 12$ ,  $\kappa = 0.2$ ,  $w_l = 0.5$ ,  $w_e = 0.5$ ,  $w_l = 0$ , and *iteration* = 50

**Fig. 5** Three steps for image segmentation. **a** Picking the reference points of each vertebra, **b** picking the center points of all vertebrae and **c** segmented data by implementing optimized geometric and calibrating parameters using the snakes algorithm

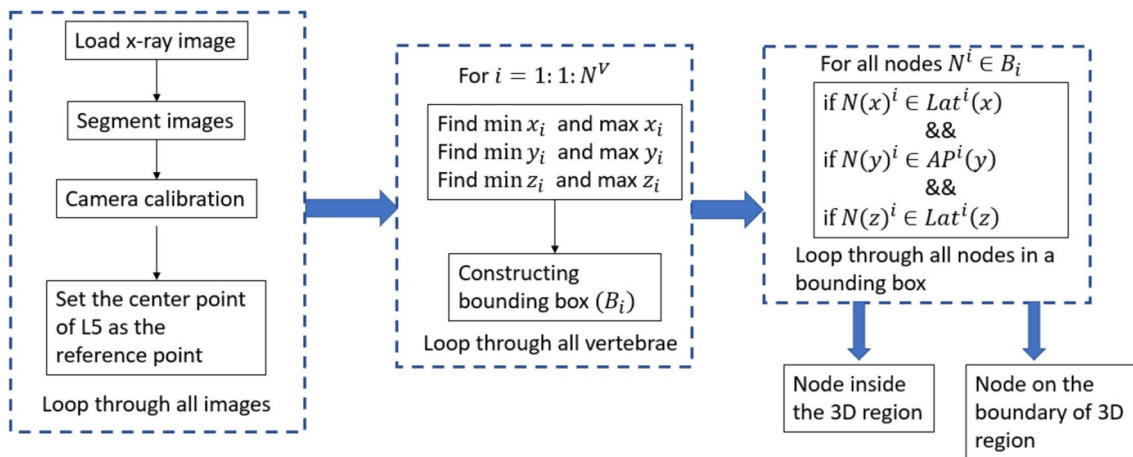


- *Reference of the coordinate system:* The coordinate system’s reference point in AP and LAT perspectives is the center point of L5 (5<sup>th</sup> lumbar vertebra).
- *Angle between AP and LAT views:* the angle between AP and LAT views is 90°.
- *Scaling criterion:* The images are taken with different scales. Images need to be scaled such that the heights of the spine in these two X-ray views are the same. It is assumed that AP view is fixed and the LAT view will be calibrated accordingly. The scaling factor between two pairs of images is calculated as

$$s = \frac{z_{\max}^{\text{AP}} - z_{\min}^{\text{AP}}}{z_{\max}^{\text{LAT}} - z_{\min}^{\text{LAT}}} \tag{4}$$

where  $z_{\max}^{\text{AP}}$  and  $z_{\min}^{\text{AP}}$  are the maximum and minimum of  $z$  coordinates of landmarks in AP, respectively. Similarly,  $z_{\max}^{\text{LAT}}$  and  $z_{\min}^{\text{LAT}}$  are the maximum and minimum of  $z$  coordinates of landmarks in LAT, respectively.

As seen in Fig. 6, the scaled data is subsequently fed into a 3D reconstruction procedure. The size of the bounding box for each vertebra is calculated by identifying the minimum and maximum in each direction, and the nodes in 3D space corresponding to that region are identified. The nodes on the boundary will also be detected using the MATLAB *convhull* function [45] (Convex hull of the 3D reconstructed region). The 3D reconstruction geometry will be obtained as well as the boundary points for every vertebrae by performing



**Fig. 6** Flowchart of 3D reconstruction of vertebrae using 2D data

the described technique and utilizing the scaled landmarks received from the X-ray images (Fig. 7).

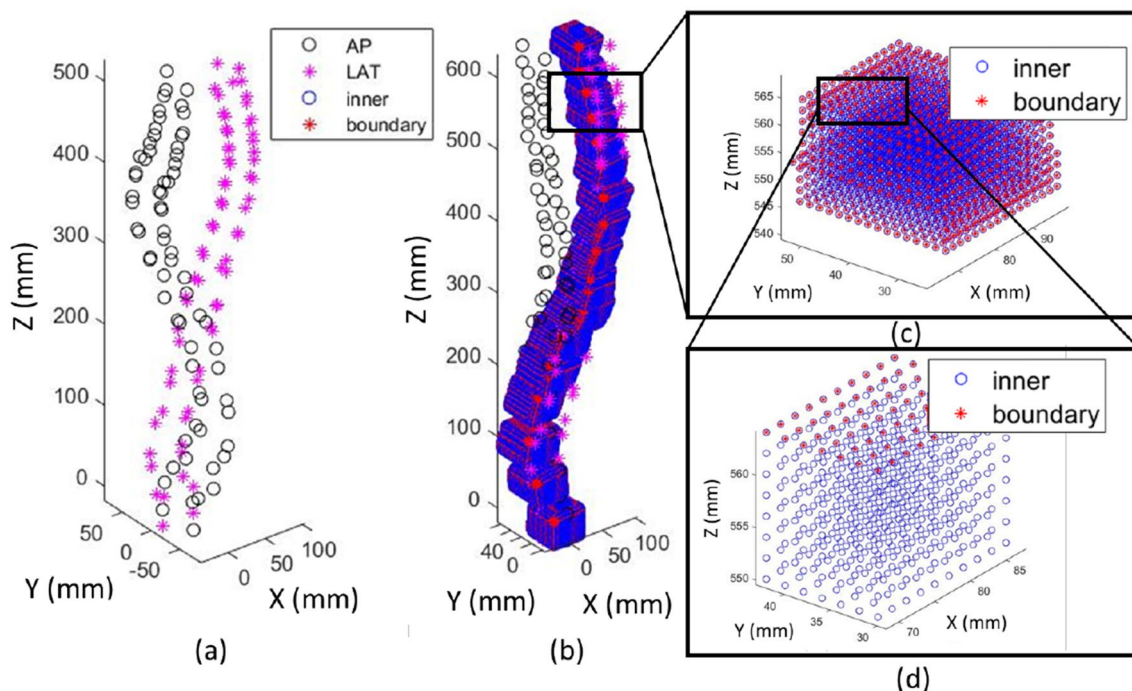
### 3.3 Volumetric mesh generation

The 3D reconstructed shape shown in Fig. 7 cannot be directly used in to generate 3D volumetric mesh of an estimated spine from the 2D views, and we use a twisting registration for the spine on the source mesh introduced in [46]. In our previous work [18], we used free-form deformation based on truncated hierarchical B-splines (THB-splines) for registration [47–49]. During registration, the detailed source geometry is taken from the atlas 3D geometry model which includes 128,205 vertices as explained in [18]. The 3D volumetric mesh composes of three components: intervertebral discs, lumbar vertebrae and thoracic vertebrae. The volumetric mesh of each vertebra is then segmented into growth plate, cortical bone and cancellous bone regions. Similarly, the volumetric mesh of each intervertebral disc is partitioned into Annulus Fibrosus and Nucleus Pulposus regions.

The employed registration method is based on the Iterative Closest Point (ICP) technique [50] which is extensively explained in Appendix B. In this method, we use the central points of each estimated vertebra and 8 corner points of the corresponding bounding box as the target points. Then we

define the point cloud as a surface and perform the nonrigid registration method in Appendix B with high stiffness i.e., high  $\gamma$  in Eq. (B9), to register the mentioned corresponded points from source mesh (the Atlas 3D geometry in [18]) to the target mesh (generated in Sect. 3.2). The stiffness term prohibits the unfavourable rotation and causes in logical twisting instead. After finding the limited number of the transformation matrix as outcome of the registration method ( $17$  (vertebra)  $\times 8$  (bounding box)  $+ 17$  (central of vertebra)  $= 153$  transformation matrices in total), each point on the source mesh among the 128,205 vertices will be transformed based on the transfer matrix of each of the 153 points which is the closest to the vertex before registration. As shown in Fig. 8a, b, the central points of the vertebra on the target spine and the source spine are depicted before and after registration respectively. Using the found transformation matrix of these 153 points, the complete volumetric spine is deformed from Fig. 8c to d which matches all the 8 landmarks of each vertebra.

Practically, the registration method allows us to generate a time-series mesh morphing of a patient's spine using the output of Sect. 3.2 where the X-ray images are serial images taken at different time steps from the spine. Through the approach introduced in Sect. 3.2, we have a low-quality time-series 3D point clouds of the spine. Here, we use each of the reconstructed spines in each time interval as

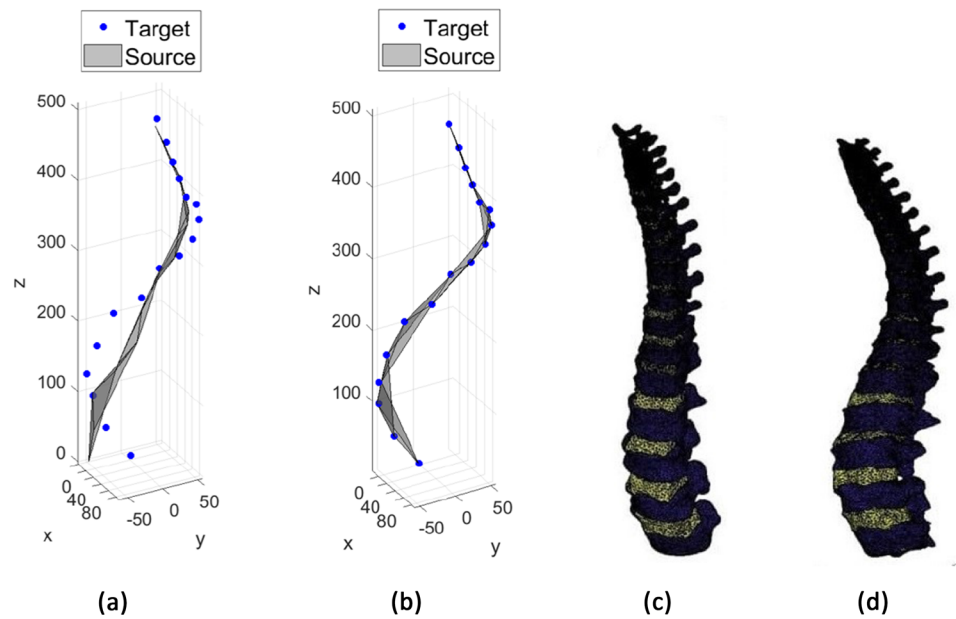


**Fig. 7** 3D reconstruction of vertebrae using 2D data. The camera parameters are calibrated before reconstruction based on the explanation in Sect. 3.2. **a** The segmented data of the AP and LAT views, **b** the 3D reconstructed geometry shaded in blue (obtained from the

multiple inner volume blue points) using the bounding box algorithm; the magenta points denote the boundary (outer) surface and the blue points denote the inner volume, **c** and **d** zoomed-in views of the inner and boundary points of the third thoracic vertebra (T3)



**Fig. 8** Volumetric mesh registration on a point cloud including reconstructed landmarks from X-ray images. Before registration: **a** center line, **c** volumetric mesh; After registration: **b** center line, **d** volumetric mesh



a template for the mesh registration method explained in Appendix B, which gives us a mesh morphing 4D (3D + time) [51, 52] data of the spine.

### 4 Bio-informed mechanistic machine learning

In this section, we propose our bio-informed mechanistic machine learning model that incorporates clinical data as well as mechanistic knowledge such as bone growth model and finite element results. We propose three machine learning models for prognosis of PSD: 3D-Clinical Neural Network; Bio-informed Clinical Neural Network; and Bio-informed Mechanistic Neural Network. The Bio-informed Clinical Neural Network is a stand-alone model that utilizes only the clinical data (i.e., X-ray images). Each of the latter two models is divided into two parts where both of them share the first part, called Center Point Predictor Neural Network. The Center Point Predictor Neural Network returns the center point of each vertebra, and this output is fed into the second part of the respective model.

The major difference between the Bio-informed Clinical Neural Network and the Bio-informed Mechanistic Neural Network is the loss function of the second part. The former uses the bone growth parameters in the loss function, whereas the latter takes both geometric features (position of landmarks and center points) and bone growth parameters into the loss function. This will be explained in detail in the following Sect. 4.3.

### 4.1 Introduction to bone growth model

One way to address the scarcity of patient-specific data is to incorporate the governing physical equation since it provides insights into the expected data. There are several aspects to consider when employing governing physical equations. The model must be validated over a large population, and some patient-specific time-dependent constants must be calibrated for each individual patient at each time step if these models are used in computational simulations as well as ML frameworks.

The underlying physical equation that governs the progression of the spine is the bone growth model, because the AIS occurs during adolescence when bone growth is at its peak [53]. It is believed that bone formation is a consequence of the stress imposed on each vertebra’s growth plate [54] located on the top and bottom of each vertebra. This assumption, however, may not be adequate to correctly depict bone growth, since other factors such as age, sex, and bone mineral density also affect bone formation. In the presented approach, we describe bone growth as follows:

$$G = A + B\bar{\sigma} \tag{5}$$

where  $G = [G_x, G_y, G_z]$  are the growth rates along three normal directions and  $\bar{\sigma}$  is the von Mises stress. There are two advantages of implementing von Mises stress. First, all stress components are taken into account and second, the definition of the stress is independent of the coordinate system. Vectors  $A$  and  $B$  are patient-specific parameters that should be calibrated for each patient, where  $A = [A_x, A_y, A_z]$  are considered as growth rate for non-scoliotic spine

and  $\mathbf{B} = [B_X, B_Y, B_Z]$  are the regulating growth model parameters.

## 4.2 Modified growth model

We aim to make the most use of all available data by exploiting medical features acquired from X-ray data, mechanistic features extracted from computational models, and addressing data scarcity while leveraging the underlying physical equation. The objective of the bio-informed mechanistic machine learning method is to reconstruct the curvature of the spine throughout time. However, the calibrated parameters generated by the system may be implemented into a computational model for monitoring and assessing bone formation on a local level.

### 4.2.1 Framework explanation and notation

The parameters of the physical growth equation are *patient-specific, time-dependent, and position-dependent*. Thus, Eq. (5) for any arbitrary landmarks on the growth plates for time  $t$  and vertebra  $v$  can be rewritten as

$$\mathbf{G}^{tv} = \mathbf{A}^{tv} + \mathbf{B}^{tv} \bar{\sigma}^{tv} \quad (6)$$

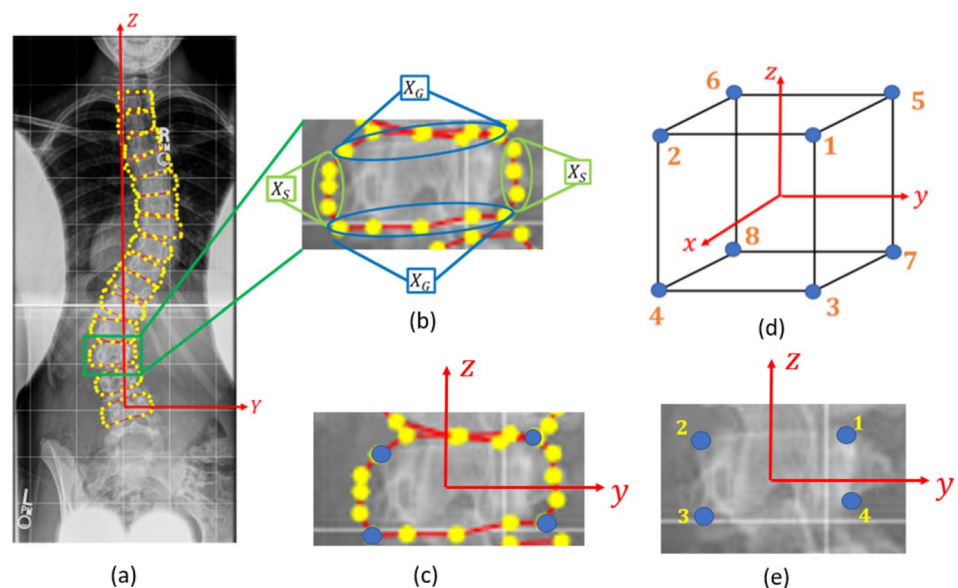
where  $\mathbf{A}^{tv} = [A_X^{tv}, A_Y^{tv}, A_Z^{tv}]$  and  $\mathbf{B}^{tv} = [B_X^{tv}, B_Y^{tv}, B_Z^{tv}]$  are patient-specific time-dependent, position-dependent parameters that should be calibrated for each patient at a given time to calculate growth rates  $\mathbf{G}^{tv} = [G_X^{tv}, G_Y^{tv}, G_Z^{tv}]$  using the von Mises stress at time  $t$  for vertebra  $v$  using the von Mises stress field from the surrogate finite element model explained in [18]. The loads, boundary conditions and material properties are the function of time  $t$  and is updated accordingly based on the age of the patient. The finite element model's output will

contain the von Mises stress that corresponds to the growth landmarks' position. Indices  $X$ ,  $Y$  and  $Z$  correspond to the global coordinate system shown in Fig. 7. In plain terms, the modified growth equation describes the directional growth of a specific vertebra of a particular patient over a span of time. The spatiotemporal parameters of the equation track the co-ordinates of the vertebra over the time. One can imagine a bounding box around a vertebra that deforms in three dimensions. The growth equation tracks the three orthonormal components of the growth through the parameters. Hence, once solved, the modified growth equation can specify the later position of the bounding box.

**Local and global coordinate systems** To obtain the growth model parameters that correspond to the growth deformation of the vertebrae without taking into account the rigid body motion caused by the movement of the bottom vertebrae, a local coordinate system is set up at the center point of each vertebra. The axes of the local coordinates have the same direction as the axes of the global coordinate system (as mentioned in Sect. 3, the center of global coordinate system is at the center point of  $L5$  (the fifth lumbar vertebra)). Figure 9a shows the global coordinate systems corresponding to the time  $t$  and the local coordinate system for the vertebra  $L3$  at the same time (Fig. 9b). The landmarks that surround each vertebra are divided into two categories: those that are positioned on the growth plates ( $X_G$ ) and those that dictate the vertebra's side shape ( $X_S$ ) as shown in Fig. 9c.

**Growth parameters** The growth landmarks with comparable motion directions (due to growth) within the same vertebra should be analyzed together to represent the growth deformation of  $X_G$  landmarks. In the first step, nodes that behave similarly in each direction will be grouped together, as indicated in Table 2.

**Fig. 9** Description of the coordinate systems and landmarks. **a** Global coordinate system in AP view, **b** labeling landmarks to growth ( $X_G$ ) and side ( $X_S$ ), **c** local coordinate system of the vertebra  $L3$  in AP view (the yellow dots are the 16 landmarks and the blue dots are the 4 corner landmarks), **d** labeling of the growth landmarks on vertebrae in local coordinate system in 3D view and **e** labeling of the growth landmarks on the  $L3$  in local coordinate system in 2D view using the 4 corner points



In the  $y$ -axis, for example, landmarks 1, 3, 5 and 7 have the same growth order (positive local  $y$ ), whereas landmarks 3, 4, 7 and 8 have the same growth direction in the  $z$ -axis (negative local  $z$ ) as visualized in Fig. 9. The growth equation of each landmark ( $j$ ) in local coordinates ( $x, y, z$ ) for time ( $t$ ) and vertebra ( $v$ ) can be written as:

$$G_j^{tv} = A_j^{tv} + B_j^{tv} \bar{\sigma}_j^{tv} \tag{7}$$

where  $A_j^{tv} = [A_{xj}^{tv}, A_{yj}^{tv}, A_{zj}^{tv}]$  and  $B_j^{tv} = [B_{xj}^{tv}, B_{yj}^{tv}, B_{zj}^{tv}]$  are patient-specific time dependent, position dependent parameters for landmark ( $j$ ) that should be calibrated for each patient at a given time for every vertebra. Indices  $x, y$  and  $z$  represent the three directions of local coordinate system shown in Fig. 9. It should be noted that  $\bar{\sigma}_j^{tv}$  is the von-Mises stress on landmark  $j$  which is independent of the coordinate system.

### 4.2.2 Calculating growth parameters

Nodes with comparable growth behavior have the same growth parameters, as listed in Table 2. These parameters are considered to be the same for each vertebra. The first step in determining growth parameters for each vertebra  $v$  is to create growth equations for all nodes as presented in Eq. (7).

The equations are rewritten to represent the clustered label in each direction, using the same growth parameter assumption as mentioned before as

$$G_{xj}^{tv} = \begin{cases} A_{xF}^{tv} + B_{xF}^{tv} \bar{\sigma}_j^{tv} & x_j^{tv} < 0, \\ A_{xB}^{tv} + B_{xB}^{tv} \bar{\sigma}_j^{tv} & x_j^{tv} \geq 0, \end{cases} \tag{8}$$

$$G_{yj}^{tv} = \begin{cases} A_{yR}^{tv} + B_{yR}^{tv} \bar{\sigma}_j^{tv} & y_j^{tv} < 0, \\ A_{yL}^{tv} + B_{yL}^{tv} \bar{\sigma}_j^{tv} & y_j^{tv} \geq 0, \end{cases} \tag{9}$$

$$G_{zj}^{tv} = \begin{cases} A_{zU}^{tv} + B_{zU}^{tv} \bar{\sigma}_j^{tv} & z_j^{tv} < 0, \\ A_{zD}^{tv} + B_{zD}^{tv} \bar{\sigma}_j^{tv} & z_j^{tv} \geq 0 \end{cases} \tag{10}$$

where  $x_j^{tv}, y_j^{tv}$  and  $z_j^{tv}$  are the coordinates in the local system of landmark  $j$  for the vertebra  $v$  at time  $t$ . The vectors  $A^{tv}$  and  $B^{tv}$  for a vertebra  $v$  at time  $t$  will be defined as

**Table 2** Landmarks with the same growth behavior are clustered into the same group. The numbers are clarified in Fig. 9d

Local growth direction	Clustered landmarks
$x$	$F = [1, 2, 3, 4]$ $B = [5, 6, 7, 8]$
$y$	$R = [1, 3, 5, 7]$ $L = [4, 2, 6, 8]$
$z$	$U = [1, 2, 5, 6]$ $D = [3, 4, 7, 8]$

$$\begin{aligned} A^{tv} &= [A_F^{tv}, A_B^{tv}, A_R^{tv}, A_L^{tv}, A_U^{tv}, A_D^{tv}], \\ B^{tv} &= [B_F^{tv}, B_B^{tv}, B_R^{tv}, B_L^{tv}, B_U^{tv}, B_D^{tv}], \end{aligned} \tag{11}$$

to define the patient-specific, time-dependent, and position-dependent scalar value of growth parameters for each vertebra. To discretize the growth parameters in each direction, the vectors

$$\begin{cases} A_x^{tv} = [A_F^{tv}, A_F^{tv}, A_F^{tv}, A_F^{tv}, A_B^{tv}, A_B^{tv}, A_B^{tv}, A_B^{tv}], \\ A_y^{tv} = [A_R^{tv}, A_L^{tv}, A_R^{tv}, A_L^{tv}, A_R^{tv}, A_L^{tv}, A_R^{tv}, A_L^{tv}], \\ A_z^{tv} = [A_U^{tv}, A_U^{tv}, A_D^{tv}, A_D^{tv}, A_U^{tv}, A_U^{tv}, A_D^{tv}, A_D^{tv}], \\ B_x^{tv} = [B_F^{tv}, B_F^{tv}, B_F^{tv}, B_F^{tv}, B_B^{tv}, B_B^{tv}, B_B^{tv}, B_B^{tv}], \\ B_y^{tv} = [B_R^{tv}, B_L^{tv}, B_R^{tv}, B_L^{tv}, B_R^{tv}, B_L^{tv}, B_R^{tv}, B_L^{tv}], \\ B_z^{tv} = [B_U^{tv}, B_U^{tv}, B_D^{tv}, B_D^{tv}, B_U^{tv}, B_U^{tv}, B_D^{tv}, B_D^{tv}], \end{cases} \tag{12}$$

are defined corresponding to the growth parameters in each direction for all the growth landmarks. In the presented frame work, the number of growth landmarks is  $X_G = 8$ , corresponding to 8 corner points shown in Fig. 9. The post-processed data of serial X-ray images (described in Sect. 3.2) are used to directly assess the growth parameters by monitoring the locations of the growth landmarks in the two consecutive images. To represent the matrix form of growth parameters for each vertebra, matrices  $\mathbb{A}$  and  $\mathbb{B}$  are developed as

$$\begin{aligned} \mathbb{A}^{tv} &= [[A_x^{tv}]^T \ [A_y^{tv}]^T \ [A_z^{tv}]^T], \quad \text{and} \\ \mathbb{B}^{tv} &= [[B_x^{tv}]^T \ [B_y^{tv}]^T \ [B_z^{tv}]^T]. \end{aligned} \tag{13}$$

The dimension of  $\mathbb{A}^{tv}$  and  $\mathbb{B}^{tv}$  are  $8 \times 3$  corresponding to 8 growth landmarks and 3 local coordinate directions ( $x, y$  and  $z$ ).

### 4.3 Spinal deformity prognosis framework

A bio-informed machine learning framework is introduced by incorporating our knowledge from clinical data (X-ray images) and the modified bone growth computational model explained in Sect. 4.2. The landmark position  $X$  is related to the features  $I$  at a given time by an unknown bio-physics equation which can be generically defined as a mapping. Note that  $I$  is referring to general features for now. It will be specified in the following sections depending on the choice of the model. The overall structure of a neural network can also be described as a mapping, i.e.:

$$\begin{cases} \text{Unknown bio-physics equation: } X = \mathcal{F}_{\text{Unknown-BioPhy}}(I) \\ \text{Neural network mapping: } X = \mathcal{F}_{\text{FFNN}}(I) \end{cases} \tag{14}$$

where  $\mathcal{F}_{\text{FFNN}}$  is the feed forward neural network (FFNN) that uses clinical and mechanistic features  $I$  as input, and

generates landmark coordinates  $\mathbf{X}$  as the output. The notation used throughout this section is defined in Table 3. To predict the position of a landmark, the input neurons would be clinical and mechanistic features and the hidden neurons would act as a multiplicative, functional decomposition of the unknown bio-physics equation that estimates the unknown function required to map input features to coordinates in the output neurons. In this project, MATLAB is used to build the FFNN and to train the neural network parameters [55].

Each neuron within every layer of a generic FFNN receives the output value from each neuron in the previous layer as input and produces a single output. This procedure is carried out for each layer. For an arbitrary number of layers and neurons per layer, the value of the  $j$ th neuron in layer  $l$  for the  $s$ th sample (either a training sample or prediction) may be written as:

$$a_{j,s}^l = \begin{cases} I_{j,s}, & \text{if } l = 1 \text{ (input layer)} \\ \mathcal{A}(b_j^l + \sum_{i=1}^{N_N^{(l-1)}} W_{ji}^{l-1} a_{i,s}^{l-1}), & \text{if } 1 < l < N_L \text{ (hidden layers)} \\ b_j^l + \sum_{i=1}^{N_N^{(l-1)}} W_{ji}^{l-1} a_{i,s}^{l-1}, & \text{if } l = N_L \text{ (output layer)} \end{cases} \quad (15)$$

where  $\mathcal{A}$  is an activation function. In the training part, this research uses *ReLU* (Rectified Linear Unit) function defined as the positive part of its argument, [56]:  $f(x) = x^+ = \max(0, x)$ , and each neuron is computed using

a different weight  $W_{ij}^{l=1}$  and bias  $b_j^{l=2}$ , where  $i$  is the neuron in the previous layer. Finally, the overall response—the predicted landmark coordinates—is given by:

$$\mathbf{X}_{j,s}^{\text{predicted}} = a_{j,s}^{N_L}. \quad (16)$$

The FFNN can learn the unknown governing bio-physics equation based on the loss function and input and output features. The loss function may be constructed to train the FFNN to offer accurate estimates while also supplying patient-specific parameters that can be considered as the network's hyperparameter. Multiple FFNN approaches are presented in the next section, each with a distinct loss function and input-output features.

### 4.3.1 Machine learning framework explanation

*3D-clinical neural network* (FFNN<sub>CL</sub>): Given the coordinates of a vertebra's landmarks at time  $t$ , the clinical neural network attempts to predict the vertebra's landmark coordinates at time  $t + \Delta t$ ; see Fig. 10. The input vectors are  $\mathbf{I}_s^t = [\bar{\mathbf{X}}_s^t, \boldsymbol{\alpha}_s^t, t, \Delta t]$ , where  $\bar{\mathbf{X}}_s^t$  indicates the landmark coordinates of sample  $s$  at time  $t$  and  $\boldsymbol{\alpha}_s^t$  denotes the global angles (described in [18]) characterizing the 3D shapes of the spine at that time. The outputs are vectors  $\mathbf{O}_s^t = [\mathbf{X}_s^{t+\Delta t}]$ , where  $\mathbf{X}_s^{t+\Delta t}$  denotes the expected 3D coordinates of the vertebra's corner points at time  $t + \Delta t$ .

---

#### 3D-Clinical neural network (FFNN<sub>CL</sub>):

---

*Input:*  $\mathbf{I}_s^t = [\bar{\mathbf{X}}_s^t, \boldsymbol{\alpha}_s^t, t_0, t]$ .

*Output:*  $\mathbf{O}_s^t = [\mathbf{X}_s^{t+\Delta t}]$ .

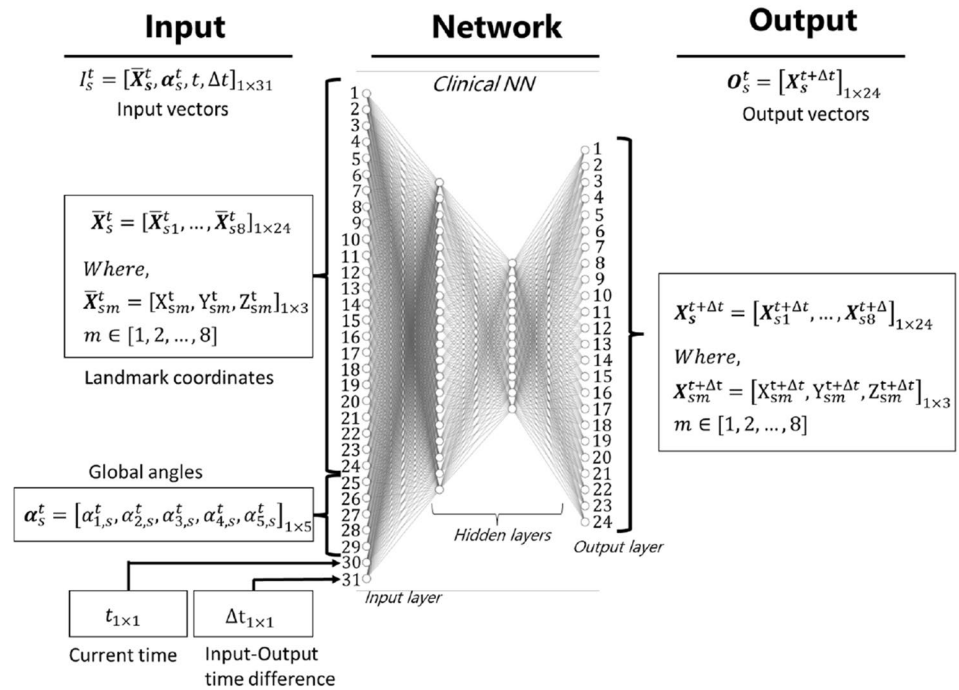
Find  $W_{ij}^l (l \in \{1, 2, 3\}), b_j^l (l \in \{2, 3, 4\})$  for each patient to minimize

$$\text{loss}_{CL}^t = \frac{1}{N_T} \frac{1}{N_m} \sum_{s=1}^{N_T} \sum_{m=1}^{N_m} \left\| \mathbf{X}_s^{t+\Delta t} - \bar{\mathbf{X}}_s^{t+\Delta t} \right\|^2, \quad (17)$$

$$\begin{aligned} \text{where } \mathbf{O}_s^t = [\mathbf{X}_s^{t+\Delta t}] &= \left[ b_q^{l=4} + \sum_{k=1}^{N_N^{[l=3]}} W_{qk}^{l=3} \mathcal{A} \left( a_{k,s}^{l=3} \right) \right]_{q=1}^{N_N^{[l=4]}}, \\ a_{k,s}^{l=3} &= b_k^{l=3} + \sum_{j=1}^{N_N^{[l=2]}} W_{kj}^{l=2} \mathcal{A} \left( a_{j,s}^{l=2} \right), \\ a_{j,s}^{l=2} &= b_j^{l=2} + \sum_{i=1}^{N_N^{[l=1]}} W_{ji}^{l=1} I_{i,s}^t, \\ \bar{\mathbf{X}}_s^{t+\Delta t} &\text{ is from clinical dataset.} \end{aligned}$$


---

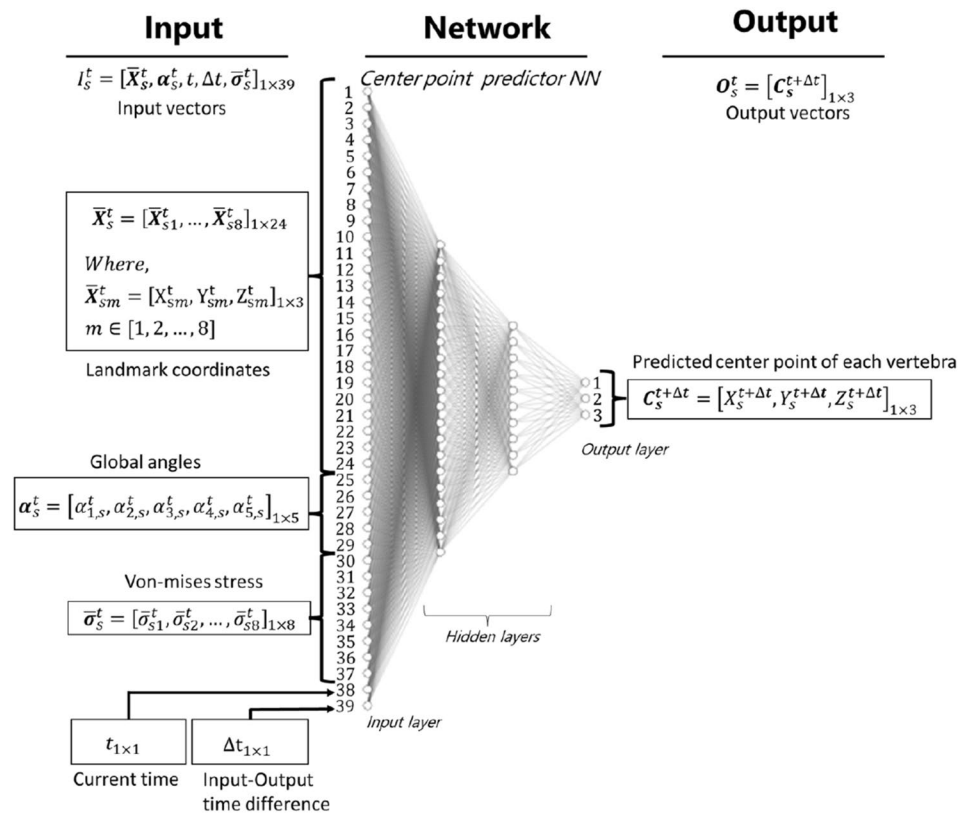
**Fig. 10** The structure of the 3D-clinical feed-forward neural network FFNN<sub>CL</sub> developed to predict the coordinates of the landmarks at time  $t + \Delta t$



The relative approximation error for this model is calculated as

$$\delta_{CL} = \frac{1}{\mathcal{H} \cdot N_T \cdot N_m} \sum_{s=1}^{N_T} \sum_{m=1}^{N_m} \|X_{sm}^{t+\Delta t} - \bar{X}_{sm}^{t+\Delta t}\|^2, \quad (18)$$

**Fig. 11** The structure of the center point prediction feed-forward neural network FFNN<sub>CR</sub> developed to predict the location of the center point of the vertebra at time  $t + \Delta t$



**Table 3** Notation table of variables used in the feed forward neural network

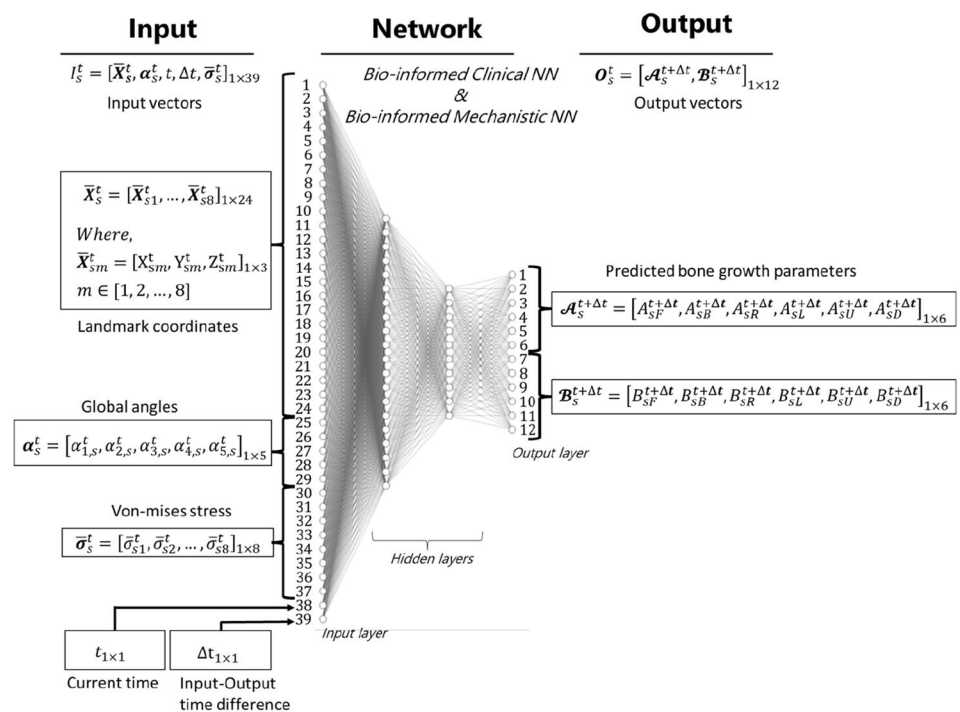
$t$	Counting index for number of time steps
$I_s^t$	Vector of input , $s = 1, \dots, N_T$
$I_{j,s}^t$	$j$ th entry of $I$ vector of input , $s = 1, \dots, N_T$
$O_s^t$	Vector of output , $s = 1, \dots, N_T$
$s$	Counting index for number of samples (training or validation, depending on context)
$m$	Counting index for number of landmarks on each vertebra
$N_m$	Number of landmarks on each vertebrae
$l$	Counting index for number of layers
$i$	Counting index for neurons in a given layer
$j$	Counting index for neurons in another layer
$N_T$	Number of training samples
$N_L$	Number of layers in the neural network
$N_N(l)$	Number of neurons in layer $l$
$a_{j,s}^l$	Neuron value for $j$ th neuron in $l$ th layer and for $s$ th sample
$W_{ji}^l$	Weight connecting the $i$ th neuron in layer $l$ to the $j$ th neuron in layer $l + 1$
$b_j^l$	Bias of the $j$ th neuron in layer $l$
$\mathcal{A}$	Activation function
$\mathcal{F}_{FFNN}$	Feedforward neural network function
$\mathcal{H}$	The height of the spine

where  $\delta_{CL}$  denotes the relative error of the clinical neural network,  $X_{sm}^{t+\Delta t}$  is the coordinates of the landmark  $m$  of vertebra  $s$  predicted by the neural network, and  $\bar{X}_{sm}^{t+\Delta t}$  is from clinical dataset.

*Center point predictor neural network (FFNN<sub>CR</sub>):* Given the set of landmarks' coordinates  $\bar{X}_s^t$ , global angles  $\alpha_s^t$ , and von Mises stress  $\bar{\sigma}_s^t$  at landmarks at time  $t$ , FFNN<sub>CR</sub> predicts the coordinates of the center of the sample at time  $t + \Delta t$ ; see Fig. 11. More formally, the input to FFNN<sub>CR</sub> is  $I_s^t = [\bar{X}_s^t, \alpha_s^t, t, \Delta t, \bar{\sigma}_s^t]$  and its output vector is  $o_{CR,s}^t = [C_s^{t+\Delta t}]$ , where  $C_s^{t+\Delta t}$  denotes the center coordinates of the sample  $s$  at time  $t + \Delta t$ . The results of this network are used in the following neural networks.

*Bio-informed clinical neural network (FFNN<sub>BC</sub>):* Using the clinical data to predict bone growth parameters, FFNN<sub>BC</sub> predicts the parameters of the physical growth equation given the set of landmark coordinates, global angles, and von Mises stress at the landmarks; see Fig. 12. Input vectors are  $I_s^t = [\bar{X}_s^t, \alpha_s^t, t, \Delta t, \bar{\sigma}_s^t]$ , where  $\bar{X}_s^t$  is the landmarks' coordinates at time  $t$ ,  $\alpha_s^t$  is the set of global angles, and  $\bar{\sigma}_s^t$  is the set of von Mises stress. The outputs of the network are vectors  $O_s^t = [A_s^{t+\Delta t}, B_s^{t+\Delta t}]$  predicting the growth model parameters in 3D for each landmark. Combining the results of FFNN<sub>BC</sub> with the center points resulted from FFNN<sub>CR</sub>, we are able to predict the coordinates of the landmarks of the sample at time  $t + \Delta t$ .

**Fig. 12** The architecture of a neural network for predicting the physical growth equation parameters. This structure is used in both FFNN<sub>BC</sub> and FFNN<sub>BM</sub> by defining two separate loss functions (one mechanistic and one non-mechanistic)



---

**Bio-informed-Clinical neural network:**

---

Step 1 (FFNN<sub>CR</sub>):

---

Input:  $\mathbf{I}_s^t = [\bar{\mathbf{X}}_s^t, \boldsymbol{\alpha}_s^t, t, \Delta t, \bar{\boldsymbol{\sigma}}_s^t]$ .

Output:  $\mathbf{O}_{CR,s}^t = [\mathbf{C}_s^{t+\Delta t}]$ .

Find  $W_{ij}^l (l \in \{1, 2, 3\}), b_j^l (l \in \{2, 3, 4\})$  for each patient to minimize

$$loss_{CR}^t = \frac{1}{N_T} \sum_{s=1}^{N_T} \left\| \mathbf{C}_s^{t+\Delta t} - \bar{\mathbf{C}}_s^{t+\Delta t} \right\|^2, \tag{19}$$

where  $\mathbf{O}_{CR,s}^t = [\mathbf{C}_s^{t+\Delta t}] = \left[ b_q^{l=4} + \sum_{k=1}^{N_N[l=3]} W_{qk}^{l=3} \mathcal{A} \left( a_{k,s}^{l=3} \right) \right]_{q=1}^{N_N[l=4]}$ ,

$$a_{k,s}^{l=3} = b_k^{l=3} + \sum_{j=1}^{N_N[l=2]} W_{kj}^{l=2} \mathcal{A} \left( a_{j,s}^{l=2} \right),$$

$$a_{j,s}^{l=2} = b_j^{l=2} + \sum_{i=1}^{N_N[l=1]} W_{ji}^{l=1} I_{i,s}^t,$$

$\bar{\mathbf{C}}_s^{t+\Delta t}$  is from clinical dataset.

---

Step 2 (FFNN<sub>BC</sub>):

---

Input:  $\mathbf{I}_s^t = [\bar{\mathbf{X}}_s^t, \boldsymbol{\alpha}_s^t, t, \Delta t, \bar{\boldsymbol{\sigma}}_s^t]$ .

Output:  $\mathbf{O}_s^t = [\mathcal{A}_s^{t+\Delta t}, \mathcal{B}_s^{t+\Delta t}]$

Find  $W_{ij}^l (l \in \{1, 2, 3\}), b_j^l (l \in \{2, 3, 4\})$  for each patient to minimize

$$loss_{BC}^t = \frac{1}{N_T} \sum_{s=1}^{N_T} \left( \left\| \mathcal{A}_s^{t+\Delta t} - \bar{\mathcal{A}}_s^{t+\Delta t} \right\|^2 + \left\| \mathcal{B}_s^{t+\Delta t} - \bar{\mathcal{B}}_s^{t+\Delta t} \right\|^2 \right), \tag{20}$$

where  $\mathbf{O}_s^t = [\mathcal{A}_s^{t+\Delta t}, \mathcal{B}_s^{t+\Delta t}] = \left[ b_q^{l=4} + \sum_{k=1}^{N_N[l=3]} W_{qk}^{l=3} \mathcal{A} \left( a_{k,s}^{l=3} \right) \right]_{q=1}^{N_N[l=4]}$ ,

$$a_{k,s}^{l=3} = b_k^{l=3} + \sum_{j=1}^{N_N[l=2]} W_{kj}^{l=2} \mathcal{A} \left( a_{j,s}^{l=2} \right),$$

$$a_{j,s}^{l=2} = b_j^{l=2} + \sum_{i=1}^{N_N[l=1]} W_{ji}^{l=1} I_{i,s}^t,$$

$\bar{\mathcal{A}}_s^{t+\Delta t}$  and  $\bar{\mathcal{B}}_s^{t+\Delta t}$  are from clinical dataset.

---

For any sample  $s$  and any landmark  $m$  on  $s$ , define the landmark  $\mathbf{X}_{BC}^{t+\Delta t}$  predicted by FFNN<sub>BC</sub> as

$$\mathbf{X}_{BC}^{t+\Delta t} = \bar{\mathbf{X}}_{BC}^t + \Delta \mathbf{C}_s^t + (\mathbb{A}_{BC}^{t+\Delta t} + \mathbb{B}_{BC}^{t+\Delta t} \boldsymbol{\sigma}_{BC}^t) \Delta t, \tag{21}$$

where  $\mathbb{A}_{BC}^{t+\Delta t}$  and  $\mathbb{B}_{BC}^{t+\Delta t}$  are obtained from  $\mathcal{A}_s^{t+\Delta t}$  and  $\mathcal{B}_s^{t+\Delta t}$ , which are the outputs of the second model,  $\bar{\mathbf{X}}_{BC}^t$  is the coordinates of landmark  $m$  of sample  $s$  at time  $t$  obtained from clinical dataset, and  $\Delta \mathbf{C}_s^t$  is the change in the center of sample  $s$  from time  $t$  to time  $t + \Delta t$ , obtained from the results of the first neural network. By this definition, this model’s relative approximation error,  $\delta_{BC}$ , is calculated as

$$\delta_{BC} = \frac{1}{\mathcal{H} \cdot N_T \cdot N_m} \sum_{s=1}^{N_T} \sum_{m=1}^{N_m} \left\| \mathbf{X}_{sm}^{t+\Delta t} - \bar{\mathbf{X}}_{sm}^{t+\Delta t} \right\|. \tag{22}$$

*Bio-informed mechanistic neural network (FFNN<sub>BM</sub>):* Similar to FFNN<sub>BC</sub>, in the first step of this model, FFNN<sub>CR</sub> predicts the coordinates of the center point of the vertebra at time  $t + \Delta t$ , followed by computing the transition vector from the center point of the sample from time  $t$  to  $t + \Delta t$ . The neural network FFNN<sub>BM</sub> then predicts the set of growth model parameters as an output using a mechanistic loss function; see Fig. 12. The input vectors are  $\mathbf{I}_s^t = [\bar{\mathbf{X}}_s^t, \boldsymbol{\alpha}_s^t, t, \Delta t, \bar{\boldsymbol{\sigma}}_s^t]$  with the same definitions for  $\bar{\mathbf{X}}_s^t, \boldsymbol{\alpha}_s^t,$  and  $\bar{\boldsymbol{\sigma}}_s^t$ . The output of FFNN<sub>CR</sub> is  $\mathbf{O}_{CR,s}^t = [\mathbf{C}_s^{t+\Delta t}]$  which remains the same and the output vectors to the second step is  $\mathbf{O}_s^t = [\mathcal{A}_s^{t+\Delta t}, \mathcal{B}_s^{t+\Delta t}]$ , with  $\mathcal{A}_s^{t+\Delta t}$  and  $\mathcal{B}_s^{t+\Delta t}$  representing the parameters of the physical growth equation.

---

**Bio-informed-Mechanistic neural network:**

---

Step 1 (FFNN<sub>CR</sub>):

---

Input:  $\mathbf{I}_s^t = [\bar{\mathbf{X}}_s^t, \boldsymbol{\alpha}_s^t, t, \Delta t, \bar{\boldsymbol{\sigma}}_s^t]$ .

Output:  $\mathbf{O}_{CR,s}^t = [\mathbf{C}_s^{t+\Delta t}]$ .

Find  $W_{ij}^l (l \in \{1, 2, 3\}), b_j^l (l \in \{2, 3, 4\})$  for each patient to minimize

$$loss_{CR}^t = \frac{1}{N_T} \sum_{s=1}^{N_T} \left\| \mathbf{C}_s^{t+\Delta t} - \bar{\mathbf{C}}_s^{t+\Delta t} \right\|^2, \tag{23}$$

where  $\mathbf{O}_{CR,s}^t = [\mathbf{C}_s^{t+\Delta t}] = \left[ b_q^{l=4} + \sum_{k=1}^{N_N[l=3]} W_{qk}^{l=3} \mathcal{A} \left( a_{k,s}^{l=3} \right) \right]_{q=1}^{N_N[l=4]}$ ,

$$a_{k,s}^{l=3} = b_k^{l=3} + \sum_{j=1}^{N_N[l=2]} W_{kj}^{l=2} \mathcal{A} \left( a_{j,s}^{l=2} \right),$$

$$a_{j,s}^{l=2} = b_j^{l=2} + \sum_{i=1}^{N_N[l=1]} W_{ji}^{l=1} I_{i,s}^t,$$

$\bar{\mathbf{C}}_s^{t+\Delta t}$  is from clinical dataset.

---

Step 2 (FFNN<sub>BM</sub>):

---

Input:  $\mathbf{I}_s^t = [\bar{\mathbf{X}}_s^t, \boldsymbol{\alpha}_s^t, t, \Delta t, \bar{\boldsymbol{\sigma}}_s^t]$ .

Output:  $\mathbf{O}_s^t = [\mathcal{A}_s^{t+\Delta t}, \mathcal{B}_s^{t+\Delta t}]$ .

Find  $W_{ij}^l (l \in \{1, 2, 3\}), b_j^l (l \in \{2, 3, 4\})$  for each patient to minimize

$$loss_{BM}^t = \frac{1}{N_T} \frac{1}{N_m} \sum_{s=1}^{N_T} \sum_{m=1}^{N_m} \left\| (\bar{\mathbf{X}}_{sm}^t + \Delta \mathbf{C}_s^t + (\mathbb{A}_{sm}^{t+\Delta t} + \mathbb{B}_{sm}^{t+\Delta t} \bar{\boldsymbol{\sigma}}_{sm}^t) \Delta t) - \bar{\mathbf{X}}_{sm}^{t+\Delta t} \right\|^2, \tag{24}$$

where  $\mathbf{O}_s^t = [\mathcal{A}_s^{t+\Delta t}, \mathcal{B}_s^{t+\Delta t}] = \left[ b_q^{l=4} + \sum_{k=1}^{N_N[l=3]} W_{qk}^{l=3} \mathcal{A} \left( a_{k,s}^{l=3} \right) \right]_{q=1}^{N_N[l=4]}$ ,

$$a_{k,s}^{l=3} = b_k^{l=3} + \sum_{j=1}^{N_N[l=2]} W_{kj}^{l=2} \mathcal{A} \left( a_{j,s}^{l=2} \right),$$

$$a_{j,s}^{l=2} = b_j^{l=2} + \sum_{i=1}^{N_N[l=1]} W_{ji}^{l=1} I_{i,s}^t,$$

$\mathbb{A}_{sm}^{t+\Delta t}$  and  $\mathbb{B}_{sm}^{t+\Delta t}$  are obtained from  $\mathcal{A}_s^{t+\Delta t}$  and  $\mathcal{B}_s^{t+\Delta t}$ , and  $\mathbf{X}_{sm}^t, \bar{\mathbf{X}}_{sm}^{t+\Delta t}$  are from clinical dataset and  $\bar{\boldsymbol{\sigma}}_{sm}^t$  is from surrogate model,  $\Delta \mathbf{C}_s^t = \mathcal{F}_{FFNNCR}(\mathbf{I}_s^t, W_{ij}^l, b_j^l)$ .

---

Similar to the FFNN<sub>BC</sub>, we define the predicted landmark  $m$  of sample  $s$ , for each  $m$  and  $s$ , as

$$\mathbf{X}_{BM}^{t+\Delta t} = \bar{\mathbf{X}}_{BM}^t + \Delta \mathbf{C}_s^t + (\mathbb{A}_{BM}^{t+\Delta t} + \mathbb{B}_{BM}^{t+\Delta t} \boldsymbol{\sigma}_{BM}^t) \Delta t. \tag{25}$$

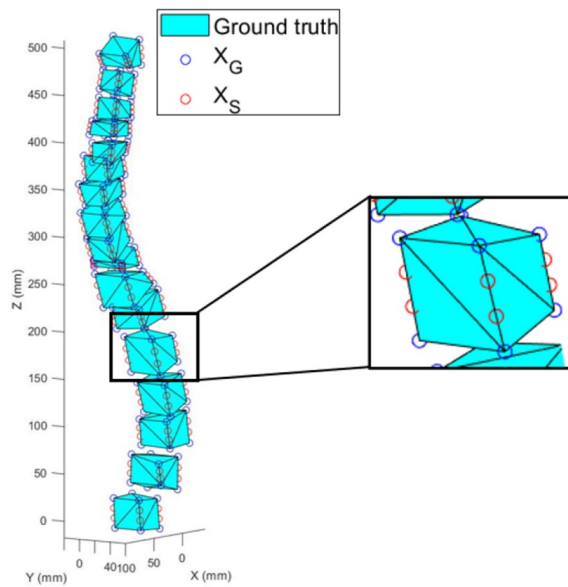
The definitions are identical to those stated in the FFNN<sub>BC</sub> formulation. The relative approximation error for FFNN<sub>BM</sub> model, represented by  $\delta_{BM}$ , is calculated using

$$\delta_{BM} = \frac{1}{\mathcal{H}.N_T.N_m} \sum_{s=1}^{N_T} \sum_{m=1}^{N_m} \left\| \mathbf{X}_{BM}^{t+\Delta t} - \bar{\mathbf{X}}_{BM}^{t+\Delta t} \right\|^2. \tag{26}$$

**4.3.2 Implementing bio-informed mechanistic machine learning frameworks**

*Data collection and feature selection* The database is comprised of clinical X-ray images. The snakes algorithm is capable of generating an infinite number of landmarks around each vertebra using X-ray images. Landmarks are classified into two types: those located on growth plates ( $X_G$ ) and those that determine the vertebral sides ( $X_S$ ), as illustrated in Fig. 9. The number of  $X_G$  landmarks on each 2D plane in this application is four, signifying the corner





**Fig. 13** Illustration of the growth landmarks and side landmarks. There are 8 growth landmarks ( $X_G = 8$ ) and 8 side landmarks ( $X_S = 8$ )

points that are expected to be on the growth plates. As a result, in the 3D framework presented ( $X_G = 8$  and  $X_S = 8$ ) as shown in Fig. 13. At  $t_0 = 124$  months, the patient’s first X-ray image is obtained. The next four X-ray images are obtained at  $t = 139, 149, 156, 168$  months and used to train the neural network. The remaining three X-ray images, collected at  $t = 160, 179, 187$  months, are utilized to compare with the neural network findings as shown in Table 5. Each X-ray image depicts the form of the patient’s spine at a specific age. Based on the explained machine learning framework explained in Sect. 4.3.1  $FFNN_{CL}$ ,  $FFNN_{CR-BC}$  and  $FFNN_{CR-BM}$  are setup as explained in Table 4. The number of training samples, 68, corresponds to the 4 training datasets (Table 5), with each dataset containing 17 vertebrae. Depending on the NN methodology, each sample has a different size. The three test datasets listed in Table 5 are represented by the number of test samples ( $17 \times 3 = 51$ ) in Table 4. It should be noted that the growth landmarks are

**Table 5** Data collection for  $FFNN_{CL}$ ,  $FFNN_{CR-BC}$  and  $FFNN_{CR-BM}$  for a single patient whose X-rays images were collected serially over time

Identification of X-ray images	Age of patient (months)
Initial X-ray image	124
Output training X-ray images	{139, 149, 156, 168}
Output testing X-ray images	{160, 179, 187}

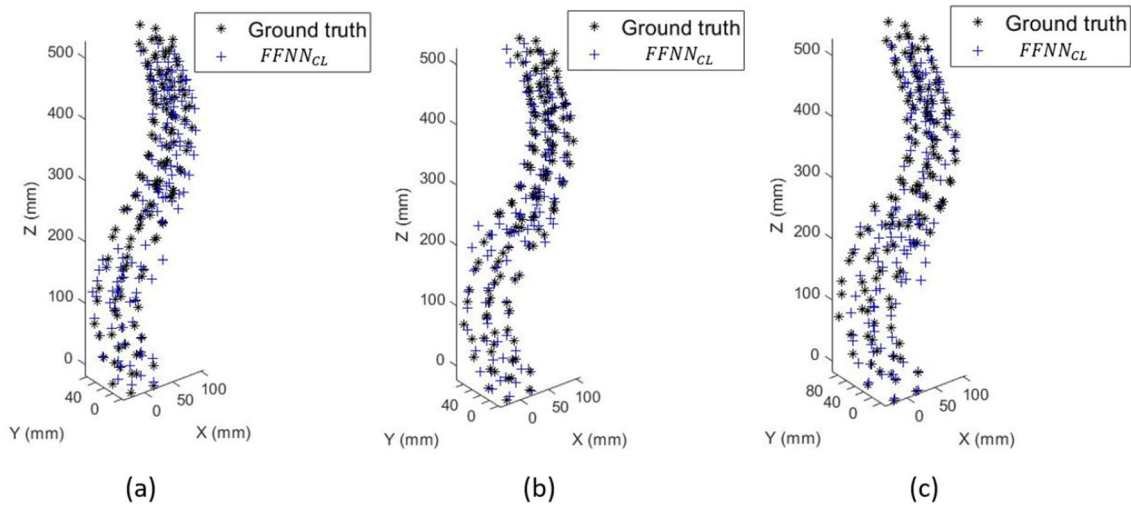
tested for each framework, and the relative approximation error indicated in Table 6 is determined for all landmarks, including growth and side landmarks. The data are normalized before being fed to the model since the range and units of the input matrix are not the same.

#### 4.4 Results and cross-validation on the data

The findings obtained from the patient described in Table 5 are presented here to illustrate the accuracy of the bio-informed mechanistic neural network prediction. The NN reconstruction process is the same as that explained in Sect. 4.3.1. A breakdown of the data used for training and testing can be seen in Table 5. Figure 14 shows a comparison of the ground truth (actual data acquired from X-ray scans) and  $FFNN_{CL}$  for the age of 160 months which is inside of the range of the training data and 179 and 187 for the outside of the range of the training data. The results are visualized for the eight corner points signifying the growth landmarks. Due to the 3D-Clinical NN’s nature, the framework fails to forecast the ground truth. This framework is built on interpolation, and when it comes to the patient-specific prediction, where the available data is insufficient, the model cannot learn from the available data, and hence it fails to predict the outcomes. Figure 15 shows the results for the  $FFNN_{CR-BC}$  framework applied on the same dataset. The results are 3D reconstructed based on the growth parameters obtained by  $FFNN_{CR-BC}$  (Fig. 15a, c and e), and the corresponding AP views are visualized in Fig. 15 b, d and f, respectively. Because the findings are recreated using the bone growth model, this framework delivers a more accurate prediction than  $FFNN_{CL}$ . The data are next examined for

**Table 4** Neural network setup for neural networks

NN Components	3D-Clinical	Bio-inf. Clinical		Bio-inf. Mechanistic	
	$FFNN_{CL}$	$FFNN_{CR}$	$FFNN_{BC}$	$FFNN_{CR}$	$FFNN_{BM}$
# training samples	68	68		68	
Size of training samples	31	39		39	
# test samples	51	51		51	
Size of test samples	24	3	12	3	12
# Hidden layers	2	2	2	2	2
Neurons in layer 1	20	20	20	20	20
Neurons in layer 2	10	10	10	10	10
Activation function	<i>ReLU</i> function				



**Fig. 14** Differences between pure data science prediction ( $FFNN_{CL}$ ) and ground truth (the results obtained by X-ray images) at age of **a** 160 months (inside of the range of the trained data), **b** 179 months (outside of the range of the trained data) and **c** 187 months (outside of

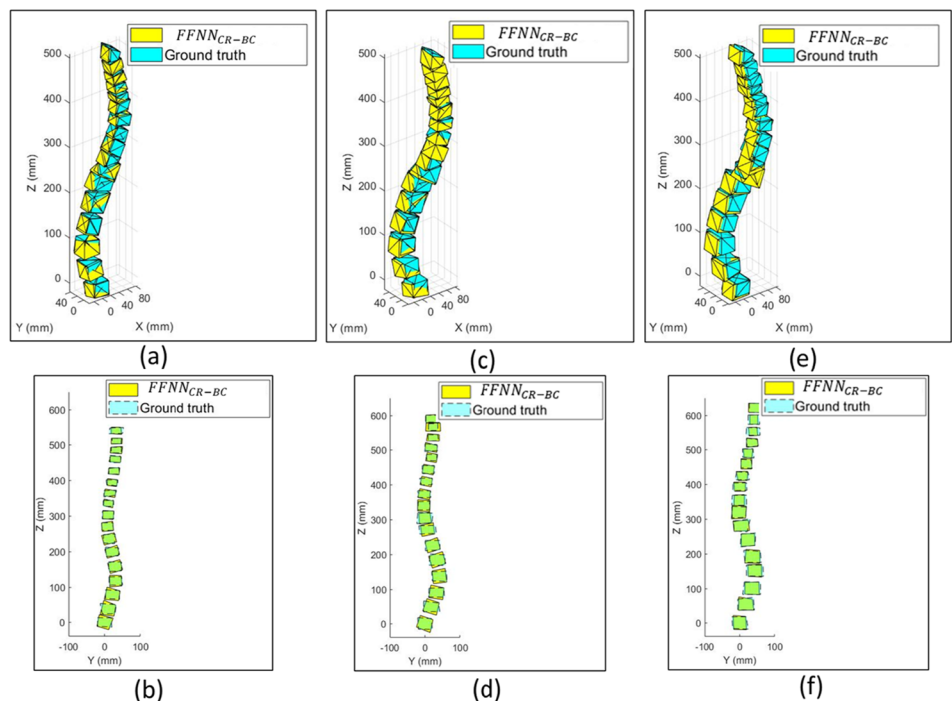
the range of the trained data). The landmarks are eight corner points of each vertebra. It is obvious that  $FFNN_{CL}$  cannot predict the ground truth

the  $FFNN_{CR-BM}$ , as illustrated in Fig. 16. Due to the implementation of the mechanistic loss function, the results are the best when compared to the  $FFNN_{CL}$  and  $FFNN_{CR-BC}$  and are compatible with the ground truth obtained from X-ray images.

A cross-validation study is carried out to see how the input and output data impact the framework error, as shown in Table 6. The term “cross validation” refers to the process

of determining how well a prediction model will work in practice [57]. We preserve the past two ages as test cases and rotate the test case inside the prior age span since we are more interested in prediction outside of the range. We can see from Table 6 that the prediction error of  $FFNN_{CR-BM}$  is always lower than that of  $FFNN_{CL}$  and  $FFNN_{CR-BC}$  and that it can predict future spine curvature with a small error. The relative errors are calculated based on the Eqs. (18), (22) and

**Fig. 15** Differences between Bio-informed Clinical prediction ( $FFNN_{CR-BC}$ ) and ground truth (the results obtained by X-ray images) at age of 160 months (inside of the range of the trained data, **a** and **b**), 179 months (outside of the range of the trained data, **c** and **d**), and 187 months (outside of the range of the trained data, **e** and **f**). Subpanels (**a**, **c**, **e**) show the 3D view and subpanels (**b**, **d**, **f**) show the 2D view on AP plane. The landmarks are eight corner points of each vertebra. It is obvious that  $FFNN_{CR-BC}$  can predict the results close to ground truth



(26) for  $FFNN_{CL}$ ,  $FFNN_{CR-BC}$  and  $FFNN_{CR-BM}$ , respectively. This observation can be explained by the fact that  $FFNN_{CL}$  operates as an interpolation function and cannot provide accurate prediction when there is insufficient training data. Since  $FFNN_{CR-BC}$  employs the bone growth equation to reconstruct the expected geometry, as explained in Eq. 20, it performs better than  $FFNN_{CL}$ . Finally,  $FFNN_{CR-BM}$  performs the best since it modifies the loss function as mentioned in Eqs. 23 and 24 while also rebuilding geometry using the biological bone growth model. Table 7 refers to the cross validation on the 2D data using the existing Mechanistic framework [58] and  $FFNN_{CR-BM}$ . As it is shown, for each testing case, the present structure is more efficient as the parameters for bone growth are calibrated taking into account the effects of time and position. The suggested  $FFNN_{CR-BM}$  enhances prediction accuracy by 40% for the inside of the range and 84.3% for the outside of the range.

### 5 Discussion and conclusion

In this study, we trained a bio-informed mechanistic deep learning model for prognosis of pediatric spinal deformity. The training data consist of clinical data and mechanistic features. The clinical data are extracted from anteroposterior and lateral views of the X-ray images using the active contour image segmentation model whose parameters are calibrated through sensitivity analysis. Next, the personalized 3D spine models are established through the mesh morphing technique which consists of registering a volumetric atlas spine model on each of the extracted clinical data. Thus, any geometry-dependant feature within the patient-specific models is detectable and trackable to calibrate the bone growth model parameters. The mechanistic features are obtained from the bone growth model which takes the stress distributions of the spine as inputs. A dynamic patient-specific bone growth model is proposed to enhance the accuracy of the model. To measure the stress distribution, a patient-specific 3D finite element model is generated based on the 2D clinical data and the 3D reconstruction algorithm. By merging medical data with a mechanistic model, the presented framework can address the problem of limited data for the patient-specific study. The proposed model is capable of predicting the spinal curve of a single patient, either inside or outside the training range. This study is unique in that it provides patient-specific, time-dependent, and position-dependent parameters that can be calibrated throughout the dataset. In addition, the proposed bio-informed deep learning network with the modified bone growth model was shown to achieve competitive or even superior performance against other state-of-the-art learning-based methods.

In conclusion, the workflow described in this article can be a useful and innovative guide for the early detection and

treatment planning of spinal illnesses such as scoliosis, lordosis, and kyphosis. Furthermore, the framework may be used for dynamic finite element analysis of various tissues at a smaller scale. It can also be potentially extended to other image-based studies, including tumor progression and cardiovascular applications. Future research will look at the same framework over a larger dataset to construct a real-time, patient-specific, optimal treatment plan based on the first patient visit. Moreover, a virtual reality application will be developed to visualize the spine in 3D and guide surgical training in orthopaedic surgery.

### Appendix A: Image segmentation using the Snakes method

Image segmentation using the Snakes method [42] is a fast and efficient technique to detect important object contours from images. The limitation of this method is that it requires initialization of the contour to be done manually. The initial contour determines the accuracy of the segmentation method. The framework involves a variational formulation in which the total energy functional consists of three main terms, namely the image energy term which attracts the contour to the salient features in the image, internal spline energy term which introduces smoothness and regularity in the evolving contour and external constrain term which aligns the contour near local minima.  $v(s) = (x(s), y(s))$  is the parametric representation of a snake where parameter  $s \in [0, 1]$ . As  $s$  changes smoothly, a closed contour on a plane is traced. The total energy functional proposed in [42] considers both image and constraint forces is given as

$$E(v(s)) = \int_0^1 E_{int}(v(s)) + E_{con}(v(s)) + E_{img}(v(s)) ds, \tag{A1}$$

where  $E_{int}(v(s))$ ,  $E_{img}(v(s))$  and  $E_{con}(v(s))$  are the energy functionals associated with internal spline energy, image force and external constraint energy, respectively. The  $E_{int}(v(s))$  term is given as

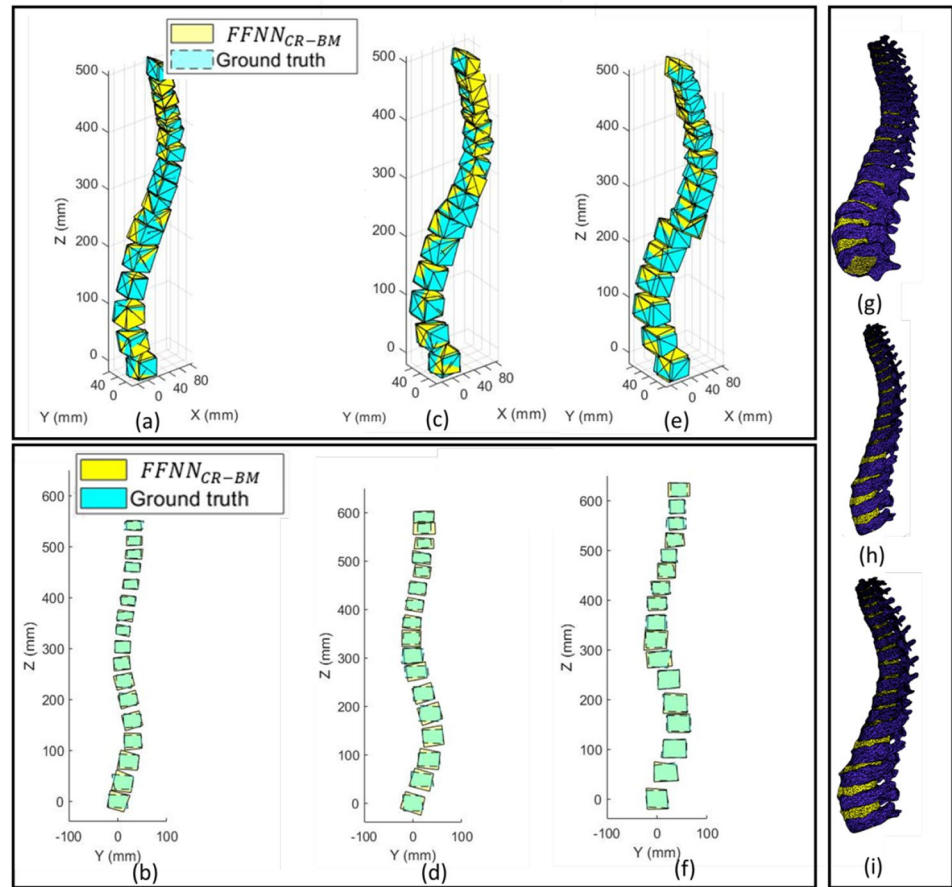
$$E_{int}(v(s)) = \alpha |v'(s)|^2 + \beta |v''(s)|^2, \tag{A2}$$

where  $\alpha$  and  $\beta$  are weights associated with the first- and second-order regularization terms which are elastic length and stiffness of the contour.  $E_{img}(v(s))$  is defined as

$$E_{img}(v(s)) = w_{line} E_{line}(v(s)) + w_{edge} E_{edge}(v(s)) + w_{term} E_{term}(v(s)), \tag{A3}$$

where  $w_{line}$ ,  $w_{edge}$  and  $w_{term}$  are the weighting coefficients associated with the energy functionals  $E_{line} = I(x, y)$ ,

**Fig. 16** Differences between Bio-informed Mechanistic predictions (FFNN<sub>CR-BM</sub>) and ground truth (results obtained by X-ray images) at age of 160 Months (inside of the range of the trained data, **a, b** and **g**), 179 months (outside of the range of the trained data, **c, d** and **h**) and 187 Months (outside of the range of the trained data, **e, f** and **i**). Subpanels (**a, c, e**) show the 3D view, subpanels (**b, d, f**) show the 2D view on AP plane and subpanels (**g, h, i**) show the 3D reconstructed detailed geometry. The landmarks are eight corner points of each vertebra. FFNN<sub>CR-BM</sub> can clearly predict outputs that are close to ground truth



$E_{\text{edge}} = -|\nabla I(x, y)|^2$  and  $E_{\text{term}} = \frac{C_{yy}C_x^2 - 2C_{xy}C_xC_y + C_{xx}C_y^2}{(C_x^2 + C_y^2)^{\frac{2}{3}}}$ ,  $I(x, y)$  is the image intensity,  $C(x, y) = G_{\sigma}(x, y) * I(x, y)$  and  $G_{\sigma}$  is a Gaussian of standard deviation  $\sigma$  [42].

### Appendix B: Point cloud registration

This section introduces the employed non-rigid ICP approach, where a concise description of the approach is given based on the conventional ICP algorithm [50].

#### B.1 The approach

In the registration process of the non-rigid ICP, the source surface  $S = (\mathcal{V}, \mathcal{E})$ , consisting of  $n$  vertices in  $\mathcal{V}$  and  $m$  edges in  $\mathcal{E}$ , is registered to the target surface  $\mathcal{T}$  step by step. Fig. 17 illustrates a step of the registration process. In the figure, the meshes are assumed to be triangular meshes, and the vertices are labeled by numbers. In this step, first, the correspondences between vertices  $v_i$  in the source surface  $S$  (green) and vertices  $u_i$  in the target surface  $\mathcal{T}$  (red) are established.

In the use of a conventional ICP method, given a point on  $S$ , the closest point on  $\mathcal{T}$  is considered as its corresponding

point [59]. Then  $v_i$  is transformed by locally affine transformations ( $X_i$ ) towards the target surface  $\mathcal{T}$  (red). The transformed source surface is  $S(X)$  (blue). This procedure iterates till an optimal stable state [60–63] is obtained.

#### B.2 3D mesh registration

Here, based on the established correspondences ( $v_i, u_i$ ), a cost function consisting of different terms is defined and then minimized with guaranteed stability, convergence, and robustness [50]. In the following, we introduce each term in the cost function first, and then we describe the optimization process based on the cost function.

For a non-rigid registration, the distance of the deformed source and the target should be minimized. Thus, a distance term is selected as the first component of the cost function to be minimised as

$$E_d = \sum_{v_i \in \mathcal{V}} w_i |X_i v_i - u_i|^2, \tag{B4}$$

where  $w_i$  is the weight of the distance term and  $X$  describes a set of transformations of displaced source vertices  $\mathcal{V}(X)$ .

**Table 6** Cross-validation study for three different neural networks. The table lists the relative approximation error of the predicted landmarks ( $X_G$  and  $X_S$  as shown in Fig. 13) of each vertebra using the dif-

ferent neural networks. Datasets are categorized by age (in months) of the tracked patient. For each trial case, the testing data are shown in blue cells and the training data are shown in white cells

NN \ Age	Prediction Error						
	139	149	156	160	168	179*	187*
<i>FFNN<sub>CL</sub></i>	<b>74.33</b>					<b>52.57</b>	<b>50.31</b>
<i>FFNN<sub>CR-BC</sub></i>	<b>1.09</b>					<b>0.10</b>	<b>0.18</b>
<i>FFNN<sub>CR-BM</sub></i>	<b>0.68</b>					<b>0.02</b>	<b>0.04</b>
<i>FFNN<sub>CL</sub></i>		<b>66.49</b>				<b>52.34</b>	<b>50.10</b>
<i>FFNN<sub>CR-BC</sub></i>		<b>0.19</b>				<b>0.05</b>	<b>0.13</b>
<i>FFNN<sub>CR-BM</sub></i>		<b>0.10</b>				<b>0.02</b>	<b>0.03</b>
<i>FFNN<sub>CL</sub></i>			<b>59.92</b>			<b>52.65</b>	<b>50.38</b>
<i>FFNN<sub>CR-BC</sub></i>			<b>0.090</b>			<b>0.091</b>	<b>0.16</b>
<i>FFNN<sub>CR-BM</sub></i>			<b>0.08</b>			<b>0.02</b>	<b>0.032</b>
<i>FFNN<sub>CL</sub></i>				<b>57.64</b>		<b>52.51</b>	<b>50.26</b>
<i>FFNN<sub>CR-BC</sub></i>				<b>0.03</b>		<b>0.08</b>	<b>0.18</b>
<i>FFNN<sub>CR-BM</sub></i>				<b>0.01</b>		<b>0.02</b>	<b>0.03</b>
<i>FFNN<sub>CL</sub></i>						<b>57.35</b>	<b>52.50</b>
<i>FFNN<sub>CR-BC</sub></i>						<b>0.03</b>	<b>0.09</b>
<i>FFNN<sub>CR-BM</sub></i>						<b>0.015</b>	<b>0.02</b>

\* Prediction at age outside of the range of trained data.

<sup>a</sup>Prediction at age outside of the range of trained data

**Table 7** Cross-validation study for two different neural networks on 2D data for AP view. The relative approximation error of the predicted landmarks ( $X_G$  and  $X_S$  as shown in Fig. 13) of each vertebra was calculated using the different neural networks. Datasets are categorized by age (in months) of the tracked patient. For each trial

case, the testing data are shown in blue cells and the training data are shown in white cells. The mechanistic framework  $FFNN_{ME}$  is borrowed from [58]. For each trial case, the bio-informed mechanistic approach ( $FFNN_{CR-BM}$ ) had better performance

NN \ Age	Prediction Error						
	139	149	156	160	168	179*	187*
<i>FFNN<sub>CR-BM</sub></i>	<b>0.14</b>					<b>0.08</b>	<b>0.03</b>
<i>FFNN<sub>ME</sub></i>	<b>0.30</b>					<b>0.12</b>	<b>0.67</b>
<i>FFNN<sub>CR-BM</sub></i>		<b>0.05</b>				<b>0.07</b>	<b>0.02</b>
<i>FFNN<sub>ME</sub></i>		<b>0.07</b>				<b>0.3</b>	<b>0.53</b>
<i>FFNN<sub>CR-BM</sub></i>			<b>0.03</b>			<b>0.06</b>	<b>0.03</b>
<i>FFNN<sub>ME</sub></i>			<b>0.04</b>			<b>0.17</b>	<b>0.20</b>
<i>FFNN<sub>CR-BM</sub></i>				<b>0.04</b>		<b>0.08</b>	<b>0.03</b>
<i>FFNN<sub>ME</sub></i>				<b>0.08</b>		<b>0.24</b>	<b>0.32</b>
<i>FFNN<sub>CR-BM</sub></i>					<b>0.04</b>	<b>0.06</b>	<b>0.03</b>
<i>FFNN<sub>ME</sub></i>					<b>0.03</b>	<b>0.24</b>	<b>0.44</b>

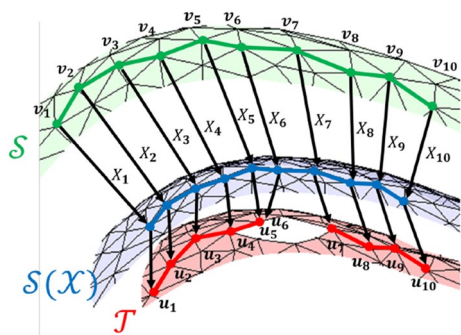
\* Prediction at age outside of the range of trained data.

<sup>a</sup>Prediction at age outside of the range of trained data

The transformation matrix  $X_i$  for each vertex in the source is a  $3 \times 4$  transformation matrix as:

$$X_i = \begin{bmatrix} r_{xx} & r_{xy} & r_{xz} & d_x \\ r_{yx} & r_{yy} & r_{yz} & d_y \\ r_{zx} & r_{zy} & r_{zz} & d_z \end{bmatrix}, \tag{B5}$$

where  $r$ , and  $d$  define all affine transformations. The transformation matrix  $X$  of all vertices is described in a  $4n \times 3$  matrix as  $X = [X_1 \dots X_n]^T$ .



**Fig. 17** Match the source surface to the target surface [64]

A canonical form of Eq. (B4) is addressed in Eq. (B6), introduced by swapping the position of transformation matrix, and correspondences  $(v_i, u_i)$ . The sparse matrix  $D$  is formed to facilitate the transformation of the source vertices with the individual transformations contained in  $X$  via matrix multiplication, and denoted as  $D = \text{diag}(v_1^T, v_2^T, \dots, v_n^T)$ . The corresponding points are also arranged as  $U = [u_1 \dots u_n]^T$  and the distance term can be derived as:

$$E_d = |W(DX - U)|_F^2 \quad (\text{B6})$$

where  $W$  is a diagonal matrix consisting of weights  $w_i$ . To regularise the deformation, an additional stiffness term is employed. Using the Frobenius norm  $|\cdot|_F$ , the stiffness term penalizes difference of the transformations of neighboring vertices, through a weighting matrix  $G = \text{diag}(1, 1, 1, \gamma)$ . We have

$$E_s = \sum_{i,j \in \mathcal{E}} |(X_i - X_j)G|_F^2 \quad (\text{B7})$$

During the deformation,  $\gamma$  is a parameter to stress differences in the skew and rotational part against the translational part of the deformation. The value of  $\gamma$  can be specified based on data units and the types of deformation [50].

Addressing the function of the stiffness term to penalise differences of transformation matrices of the neighboring vertices, the node-arc incidence matrix  $M$  (e.g. Dekker [65]) of the template mesh topology is employed to convert the stiffness term functional into a matrix form. As the matrix is fixed for directed graphs, the construction is one row for each edge of the mesh and one column per vertex. To establish the node-arc incidence matrix of the source topology, the indices (i.e. the subscripts) of edges and vertices are addressed, for any edge of  $r$  which is connected to vertices  $(i, j)$ , in  $r^{\text{th}}$  row of  $M$ , and the nonzero entries are  $M_{ri} = -1$  and  $M_{rj} = 1$ . Therefore, we formulate the stiffness term as

$$E_s = |(M \otimes G)X|_F^2 \quad (\text{B8})$$

Briefly, the Amberg's method accounts for an optimal step with non-rigid ICP approach being capable to employ different regularisations, while they are using a range of lowering stiffness parameter. Thus, the cost function of Eq. (B6) and Eq. (B8) are changed to:

$$E(X) = \left| \begin{bmatrix} \gamma M \otimes G \\ WD \end{bmatrix} X - \begin{bmatrix} 0 \\ WU \end{bmatrix} \right|_F^2 \quad (\text{B9})$$

**Acknowledgements** We would like to thank the Division of Orthopaedic Surgery and Sports Medicine at the Ann and Robert H. Lurie Children's Hospital of Chicago for their collaboration on this project, which was made possible by a philanthropic grant. W. K. Liu would like to acknowledge the support of NSF CMMI-1762035. M. Tajdari would like to thank Madeleine Handwork for her assistance in data processing. A. Pawar and Y. J. Zhang were supported in part by the NSF grant CMMI-1953323. Moreover, F. Tajdari is partly supported by the Dutch NWO Next UPPS under Grant 15470 - Integrated Design Methodology for Ultra Personalised Products and Services Project.

## Declarations

**Conflict of interest** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

1. Wiggins GC, Shaffrey CI, Abel MF, Menezes AH (2003) Pediatric spinal deformities. *Neurosurg Focus* 14(1):1–14
2. Konieczny MR, Senyurt H, Krauspe R (2013) Epidemiology of adolescent idiopathic scoliosis. *J Child Orthop* 7(1):3–9
3. Girdler S, Cho B, Mikhail CM, Cheung ZB, Maza N, Cho SK-W (2020) Emerging techniques in diagnostic imaging for idiopathic scoliosis in children and adolescents: A review of the literature. *World Neurosurg* 136:128–135
4. Knez D, Nahle IS, Vrtovec T, Parent S, Kadoury S (2019) Computer-assisted pedicle screw trajectory planning using CT-inferred bone density: a demonstration against surgical outcomes. *Med Phys* 46(8):3543–3554
5. Mischler D, Windolf M, Gueorguiev B, Nijs S, Varga P (2020) Computational optimisation of screw orientations for improved locking plate fixation of proximal humerus fractures. *J Orthop Transl* 25:96–104
6. Goerres J, Uneri A, De Silva T, Ketcha M, Reaungamornrat S, Jacobson M, Vogt S, Kleinszig G, Osgood G, Wolinsky J et al (2017) Spinal pedicle screw planning using deformable atlas registration. *Phys Med Biol* 62(7):2871
7. Müller F, Roner S, Liebmann F, Spirig JM, Fürnstahl P, Farshad M (2020) Augmented reality navigation for spinal pedicle screw instrumentation using intraoperative 3D imaging. *Spine J* 20(4):621–628
8. Liebmann F, Roner S, von Atzigen M, Wanivenhaus F, Neuhaus C, Spirig J, Scaramuzza D, Sutter R, Snedeker J, Farshad M et al (2020) Registration made easy—standalone orthopedic navigation with hololens. *arXiv preprint arXiv:2001.06209*
9. Sarkalkan N, Weinans H, Zadpoor AA (2014) Statistical shape and appearance models of bones. *Bone* 60:129–140
10. Campbell JQ, Petrella AJ (2015) An automated method for landmark identification and finite-element modeling of the lumbar spine. *IEEE Trans Biomed Eng* 62(11):2709–2716

11. Andrew J, DivyaVarshini M, Barjo P, Tigga I (2020) Spine magnetic resonance image segmentation using deep learning techniques. In: 6th International conference on advanced computing and communication systems (ICACCS), IEEE, pp 945–950
12. Li R, Niu K, Wu D, Vander Poorten E (2020) A framework of real-time freehand ultrasound reconstruction based on deep learning for spine surgery. In: 10th Conference on new technologies for computer and robot assisted surgery, 28 September 2020 to 30 September, Barcelona, Spain
13. Baum T, Bauer JS, Klinder T, Dobritz M, Rummeny EJ, Noël PB, Lorenz C (2014) Automatic detection of osteoporotic vertebral fractures in routine thoracic and abdominal MDCT. *Eur Radiol* 24(4):872–880 (Springer)
14. Tajdari F, Golgouneh A, Ghaffari A, Khodayari A, Kamali A, Hosseinkhani N (2021) Simultaneous intelligent anticipation and control of follower vehicle observing exiting lane changer. *IEEE Trans Veh Technol* 70(9):8567–8577
15. Tajdari F, Ghaffari A, Khodayari A, Kamali A, Zhilakzadeh N, Ebrahimi N (2019) Fuzzy control of anticipation and evaluation behaviour in real traffic flow. In: 2019 7th International conference on robotics and mechatronics (ICRoM), IEEE, p 248–253
16. Tajdari F, Toulkani NE, Nourimand M (2020) Intelligent architecture for car-following behaviour observing lane-changer: Modeling and control. In: 2020 10th International conference on computer and knowledge engineering (ICCKE), IEEE, p 579–584
17. Raissi M, Perdikaris P, Karniadakis GE (2019) Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J Comput Phys* 378:686–707 (Elsevier)
18. Tajdari M, Pawar A, Li H, Tajdari F, Maqsood A, Cleary E, Saha S, Zhang YJ, Sarwark JF, Liu WK (2021) Image-based modelling for adolescent idiopathic scoliosis: mechanistic machine learning analysis and prediction. *Comput Methods Appl Mech Eng*. 374:113590
19. Roberts M, Pacheco E, Mohankumar R, Cootes T, Adams J (2010) Detection of vertebral fractures in DXA VFA images using statistical models of appearance and a semi-automatic segmentation. *Osteoporos Int* 21(12):2037–2046
20. Al Arif SMR, Gundry M, Knapp K, Slabaugh G (2016) Improving an active shape model with random classification forest for segmentation of cervical vertebrae. In: International workshop on computational methods and clinical applications for spine imaging, Springer, p 3–15
21. Cootes TF (2017) Fully automatic localisation of vertebrae in CT images using random forest regression voting. In: Computational methods and clinical applications for spine imaging: 4th international workshop and challenge, CSI 2016, held in conjunction with MICCAI 2016, Athens, Greece, October 17, 2016, Springer, Revised Selected Papers, vol 10182, p 51
22. Bromiley P, Adams J, Cootes T (2015) Localisation of vertebrae on DXA images using constrained local models with random forest regression voting. In: Recent advances in computational methods and clinical applications for spine imaging, Springer, p 159–171
23. Roberts MG, Cootes TF, Adams JE (2012) Automatic location of vertebrae on DXA images using random forest regression. In: International conference on medical image computing and computer-assisted intervention, Springer, p 361–368
24. Cootes TF, Taylor CJ, Cooper DH, Graham J (1995) Active shape models-their training and application. *Comput Vis Image Underst* 61(1):38–59
25. Lamecker H, Wenckebach TH, Hege H-C (2006) Atlas-based 3D-shape reconstruction from X-ray images. In: 18th International conference on pattern recognition (ICPR '06), IEEE, vol 1, p 371–374
26. Fotsin TJT, Vázquez C, Cresson T, De Guise J (2019) Shape, pose and density statistical model for 3D reconstruction of articulated structures from X-ray images. In: 2019 41st Annual international conference of the IEEE engineering in medicine and biology society (EMBC), IEEE, p 2748–2751
27. Ehlike M, Ramm H, Lamecker H, Hege H-C, Zachow S (2013) Fast generation of virtual X-ray images for reconstruction of 3D anatomy. *IEEE Trans Visual Comput Graph* 19(12):2673–2682
28. Kim H, Lee K, Lee D, Baek N (2019) 3D reconstruction of leg bones from X-ray images using CNN-based feature analysis. In: International conference on information and communication technology convergence (ICTC), IEEE, p 669–672
29. Reyneke CJF, Lüthi M, Burdin V, Douglas TS, Vetter T, Mutsvangwa TE (2018) Review of 2-D/3-D reconstruction using statistical shape and intensity models and X-ray image synthesis: toward a unified framework. *IEEE Rev Biomed Eng* 12:269–286
30. Deschênes S, Charron G, Beaudoin G, Labelle H, Dubois J, Miron M-C, Parent S (2010) Diagnostic imaging of spinal deformities: reducing patients radiation dose with a new slot-scanning X-ray imager. *Spine* 35(9):989–994
31. Le Bras A, Laporte S, Mitton D, De Guise J, Skalli W (2003) Three-dimensional (3D) detailed reconstruction of human vertebrae from low-dose digital stereoradiography. *Eur J Orthop Surg Traumatol* 13(2):57–62
32. Dubouset J, Charpak G, Dorion I, Skalli W, Lavaste F, Deguise J, Kalifa F, Ferey S (2005) A new 2D and 3D imaging approach to musculoskeletal physiology and pathology with low-dose radiation and the standing position: the EOS system. *Bulletin de l'Academie Nationale de Medecine* 189(2):287–97
33. Bashkuev M, Reitmaier S, Schmidt H (2018) Effect of disc degeneration on the mechanical behavior of the human lumbar spine: a probabilistic finite element study. *Spine J* 18(10):1910–1920
34. Bashkuev M, Reitmaier S, Schmidt H (2020) Relationship between intervertebral disc and facet joint degeneration: a probabilistic finite element model study. *J Biomech* 102
35. Bah MT, Nair PB, Browne M (2009) Mesh morphing for finite element analysis of implant positioning in cementless total hip replacements. *Med Eng Phys* 31(10):1235–1243
36. Pasha S, Shah S, Newton P, Group HS et al (2021) Machine learning predicts the 3D outcomes of adolescent idiopathic scoliosis surgery using patient–surgeon specific parameters. *Spine* 46(9):579–587
37. Peng L, Lan L, Xiu P, Zhang G, Hu B, Yang X, Song Y, Yang X, Gu Y, Yang R et al (2020) Prediction of proximal junctional kyphosis after posterior scoliosis surgery with machine learning in the Lenke 5 adolescent idiopathic scoliosis patient. *Front Bioeng Biotechnol* 8:1–10
38. Liang R, Yip J, To K-TM, Fan Y (2021) Machine learning approaches to predict scoliosis. In: International conference on applied human factors and ergonomics, Springer, p 116–121
39. Cho J-S, Cho Y-S, Moon S-B, Kim M-J, Lee HD, Lee SY, Ji Y-H, Park Y-S, Han C-S, Jang S-H (2018) Scoliosis screening through a machine learning based gait analysis test. *Int J Precis Eng Manuf* 19(12):1861–1872
40. Saha S, Gan Z, Cheng L, Gao J, Kafka OL, Xie X, Li H, Tajdari M, Kim HA, Liu WK (2021) Hierarchical deep learning neural network (HIDENN): an artificial intelligence (AI) framework for computational science and engineering. *Comput Methods Appl Mech Eng* 373
41. Zhang YJ (2016) Geometric modeling and mesh generation from scanned images. Chapman and Hall/CRC, New York
42. Kass M, Witkin A, Terzopoulos D (1988) Snakes: active contour models. *Int J Comput Vis* 1(4):321–331
43. Tajdari F, Roncoli C, Papageorgiou M (2020) Feedback-based ramp metering and lane-changing control with connected and automated vehicles. *IEEE Trans Intell Transport Syst* 1–13

44. Tajdari F, Ebrahimi Toulkani N (2021) Implementation and intelligent gain tuning feedback-based optimal torque control of a rotary parallel robot. *J Vib Control* 1–18
45. MATLAB Convhull Function (2020) The MathWorks. Natick, MA, USA
46. Tajdari M, Tajdari F, Pawar A, Zhang J, Liu W (2021) 2D to 3D volumetric reconstruction of human spine for diagnosis and prognosis of spinal deformities. In: Conference: 16th US national congress on computational mechanics
47. Pawar A, Zhang Y, Jia Y, Wei X, Rabczuk T, Chan CL, Anitescu C (2016) Adaptive FEM-based nonrigid image registration using truncated hierarchical B-splines. *Comput Math Appl* 72(8):2028–2040
48. Pawar A, Zhang YJ, Anitescu C, Jia Y, Rabczuk T (2018) DTHB3D-Reg: dynamic truncated hierarchical B-spline based 3D nonrigid image registration. *Commun Comput Phys* 23(3):877–898
49. Pawar A, Zhang YJ, Anitescu C, Rabczuk T (2019) Joint image segmentation and registration based on a dynamic level set approach using truncated hierarchical B-splines. *Comput Math Appl* 78:3250–3267
50. Amberg B, Romdhani S, Vetter T (2007) Optimal step nonrigid ICP algorithms for surface registration. In: IEEE Conference on computer vision and pattern recognition, p 1–8
51. Tajdari F, Kwa F, Versteegh C, Huysmans T, Song Y. Dynamic 3D mesh reconstruction based on nonrigid iterative closest-farthest points registration. In: International design engineering technical conferences and computers and information in engineering conference, vol 2022, p 1–9
52. Tajdari F, Eijck C, Kwa F, Versteegh C, Huysmans T, Song Y (2022) Optimal position of cameras design in a 4D foot scanner. In: International design engineering technical conferences and computers and information in engineering conference, vol 2022, p 1–9
53. Cheung JPY, Cheung PWH, Samartzis D, Luk KD-K (2018) Curve progression in adolescent idiopathic scoliosis does not match skeletal growth. *Clin Orthop Relat Res* 476(2):429
54. Stokes I (2002) Mechanical effects on skeletal growth. *J Musculoskelet Neuron Interact* 2(3):277–280
55. MATLAB Deep Learning Toolbox (2018) The MathWorks. Natick, MA, USA
56. Agarap AF (2018) Deep learning using rectified linear units (RELU). arXiv preprint [arXiv:1803.08375](https://arxiv.org/abs/1803.08375)
57. Stone M (1974) Cross-validatory choice and assessment of statistical predictions. *J R Stat Soc Ser B (Methodol)* 36(2):111–133
58. Tajdari F, Toulkani NE, Zhilakzadeh N (2020) Intelligent optimal feed-back torque control of a 6dof surgical rotary robot. In: 2020 11th Power electronics, drive systems, and technologies conference (PEDSTC), IEEE, p 1–6
59. Yang Y, Yuan T, Huysmans T, Elkhuzen WS, Tajdari F, Song Y (2021) Posture-invariant three dimensional human hand statistical shape model. *J Comput Inf Sci Eng* 21(3)
60. Tajdari F, Tajdari M, Rezaei A (2021) Discrete time delay feedback control of stewart platform with intelligent optimizer weight tuner. In: 2021 IEEE international conference on robotics and automation (ICRA), IEEE, p 12701–12707
61. Tajdari F, Toulkani NE, Zhilakzadeh N (2020) Semi-real evaluation, and adaptive control of a 6dof surgical robot. In: 2020 11th Power electronics, drive systems, and technologies conference (PEDSTC), IEEE, p 1–6
62. Tajdari F, Roncoli C (2021) Adaptive traffic control at motorway bottlenecks with time-varying fundamental diagram. *IFAC-PapersOnLine* 54(2):271–277
63. Tajdari F, Roncoli C (2022) Online set-point estimation for feedback-based traffic control applications. arXiv preprint [arXiv:2207.13467](https://arxiv.org/abs/2207.13467)
64. Tajdari F, Huysmans T, Yang Y, Song Y (2022) Feature preserving non-rigid iterative weighted closest point and semi-curvature registration. *IEEE Trans Image Process* 31:1841–1856
65. Dekker M (1986) *Mathematical programming*. CRC, May 4

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.