Scoliosis analysis via segmentation using Attention Unet and curvature definition with B-Splines

Capstone Project Phase A

Students: Dvir Hayat, Rafael Abbassov

Advisor: Prof. Miri Weiss Cohen

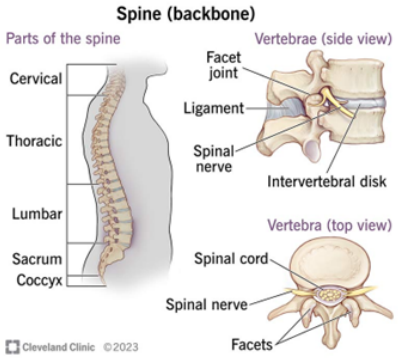
Table of Contents

Abstract. Scoliosis is a medical condition characterized by an abnormal curvature of the spine, commonly measured by the Cobb Angle.Accurate measurement of this angle and a comprehensive understanding of the spine's curvature are essential for precise treatment, necessitating high-precision segmentation and detailed description.
The project proposes a three-stage approach using spinal imaging scans: first, the vertebrae are detected using YOLOv8, then they are segmented using Attention U-Net, and finally, they are represented using a corresponding B-Spline.
In spinal imaging, YOLOv8 is a detection that is critical to identifying vertebrae accurately. Also, YOLOv8 is capable of handling complex scenes, ensuring reliable detection in even the most challenging of medical imaging environments. The Attention U-Net enhances segmentation by incorporating attention mechanisms that focus on the most relevant features of spinal images. This allows the model to effectively capture fine details and differentiate between adjacent vertebrae. Using B-Spline representations of segmented vertebrae, it is possible to model the spinal shape with great precision, even capturing subtle deviations. As a result of this detailed representation, effective treatment strategies can be devised and the progression of scoliosis can be monitored.

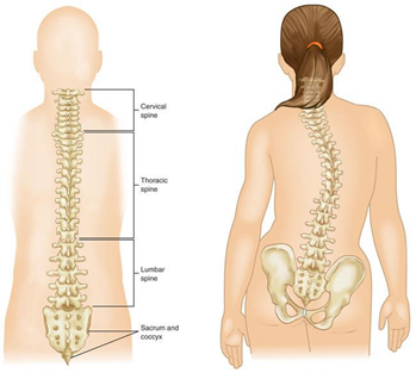
Keywords: Scoliosis, Spine Segmentation, B-Splines, Attention U-Net, YOLOv8

1 Introduction

The back bones form the vertebral column, which is divided into five sections: the cervical, thoracic, lumbar vertebrae, the sacrum, and the coccyx. Together, these 33 vertebrae support movement and posture while also providing protection and structural support [1].
We will elaborate about the three largest sections of the spine - The cervical, thoracic, and lumbar spine sections work together to protect the spinal cord, support the body, and enable movement[9]. The cervical spine, located in the neck region supports and cushions the head and neck. It allows for a wide range of motion, including rotation, while simultaneously safeguarding the spinal cord from injury[9]. The thoracic spine, situated in the mid-back, is primarily responsible for bearing significant loads from the upper body. It contributes to maintaining posture and stability in the trunk. Additionally, it is connected to the rib cage, providing protection for the vital organs within the chest, such as the heart and lungs, and plays a major role in maintaining the body's overall safety and function.
The lumbar spine, located in the lower back, is designed to provide maximum stability, supporting the heavy loads carried by the upper body while allowing mobility of the trunk relative to the hips/pelvis. It also plays a major part in bending and twisting movements. Scoliosis is a spine deformity medical condition



(a) Spine structure side view with vertebrae top and side view.[4]



(b) Normal spine anatomy and Idiopathic scoliosis in children.[13]

which is diagnosed through measuring spinal deviation on standing coronal plane radiographs [[20]]. Diagnosis usually measured through measurement of the Cobb angle. The Cobb angle is measured when 2 vertebrae are selected as the ones whose endplates are most tilted towards each other. Then, Lines are then drawn along the endplates, and the angle between the two lines, where they intersect, is measured. Scoliosis is defined as a lateral spinal curvature with a Cobb angle of $>10°$.[8][14] There are three types of Scoliosis: The most common is "Idiopathic

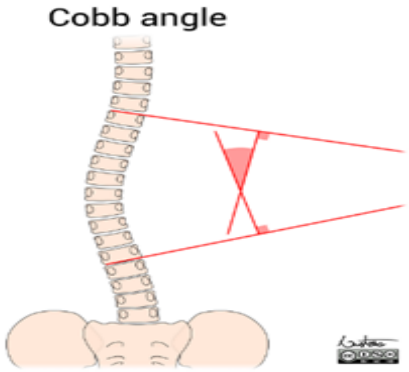


Fig. 2: Cobb angle measurement method for scoliosis[14]

scoliosis", where the main reason is uncertain however researchers believe that the condition probably stems from a variety of genetic and environmental factors, including abnormal muscle growth, hormonal issues, genetic influence and sometimes as symptom of a problematic nervous system.[23][5] The second one is "Congenital scoliosis" are rarer and often associated with pregnancy during the 5th and the 8th week, when the spine of the embryo is being developed and the bones don't form as they should. This type of scoliosis is mostly detected at the time of parental ultrasound.[5][22] The third one is "Neuromuscular sco-

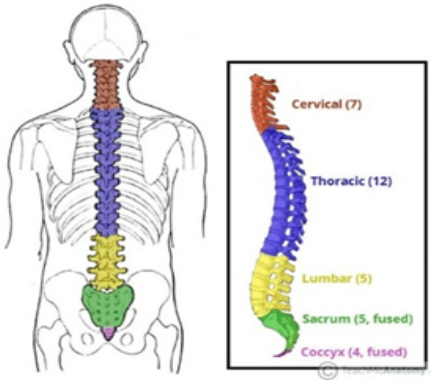


Fig. 3: The spine divided into cervical, thoracic, lumbar, sacrum, and coccyx regions.[22]

liosis," and it occurs due to problems in the muscles and nerves supporting the spine, which can be caused by trauma or muscle disease. It commonly affects posture, mobility, and can lead to respiratory or cardiovascular complications as the curve progresses [5]. Understanding the patient's structure and status of the vertebral column is crucial for diagnosing and treating spine-related disorders, particularly scoliosis, which remains a significant global health concern. Our goal is to improve the current segmentation existing abilities through deep learning methods and achieve higher accuracy that will enable medical personnel

to treat patients with more specific treatment that will hopefully eventually lead to better recovery.

2 Related work

Saeed et al. [18] suggests a segmentation approach based on the U-Net with additional network features: adding CHASPP and residual blocks to the encoder part and adding the attention module to the decoder part. U-Net is one of the most widely used models for biomedical image segmentation, named for its distinctive U-shaped architecture. It is composed of two main components: an encoder and a decoder:
Encoder: Responsible for extracting features from the input dataset.
Decoder: Utilized to reconstruct and predict the segmented mask. The model operates as a fully convolutional network, effectively capturing both localization and contextual features from the data. On the first hand The Encoder use max-

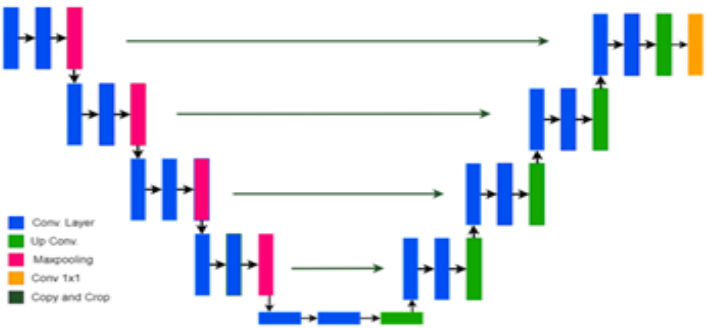


Fig. 4: U-Net model's architecture for medical image segmentation

pooling layers with stride of two. In each reduce the convolutional layers increase filters to extract features. On the second hand the Decoder up samples the image while reducing the number of filters. Skip connections that preserve the losses of important information from the previous layers get passed to the decoder. The model uses an upgraded version of ASPP which is a module used for semantic segmentation in which a feature layer can be resampled with different convolution rates. this allows the module to preserve more important information. **C**ascaded **H**ierarchical **A**trous **S**patial **P**yramid **P**ooling which is new version of **ASPP** that's created with intention to increase the number of sampling points within a receptive field. CHASPP is used in the encoder part of the proposed model for improving feature extraction. The local and the global features are extracted by the CHASPP, improving the model's performance from regular U-Net segmentation The experimental results for the CHASPPRAU-Net and model were evaluated on the VerSe 2020 and VerSe 2019 datasets for spine segmentation and

vertebrae recognition. The study included various model evaluations to measure its accurateness with and without data augmentation through various normalization methods. The results showed a slight improvement to the model with Data augmentation with the most successful normalization type a Dice score of 94.58% on VerSe2019 and 93.72% on VerSe2020 datasets. The evaluation metrics of the model (Tables below) present great performance and high accuracy for various data normalization options. Cheng et. al [3] propose a slightly differ-

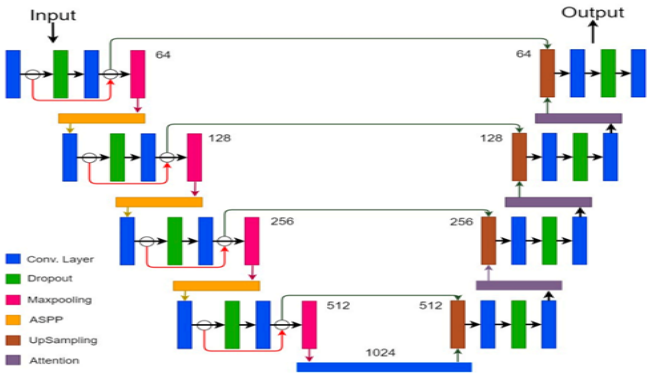


Fig. 5: CHASPP-Net improved model architecture

Dataset	Normalization	DSC (%)	IoU (%)	Precision (%)	Recall (%)
VerSe 2020	−1000 to 800	90.45	91.48	94.37	95.07
	Zero mean	89.91	90.34	95.16	94.32
	0 to 1	90.18	94.01	93.41	94.08
	−1 to 1	88.75	92.17	91.62	91.43
VerSe 2019	−1000 to 800	91.63	90.89	95.84	94.77
	Zero mean	88.42	89.37	93.25	92.98
	0 to 1	89.79	90.48	94.62	93.10
	−1 to 1	90.51	89.17	92.48	91.79

Table 1: Results of CHASPPRAU-Net for spine segmentation on VerSe 2020/2019 datasets with image normalization.

Dataset	Normalization	DSC (%)	IoU (%)	Precision (%)	Recall (%)
VerSe 2020	−1000 to 800	93.72	92.25	97.04	96.87
	Zero mean	92.63	93.14	96.13	94.53
	0 to 1	90.59	90.28	93.08	92.48
	−1 to 1	93.14	94.60	94.05	93.75
VerSe 2019	−1000 to 800	94.58	95.93	98.71	97.10
	Zero mean	92.49	91.68	95.62	95.78
	0 to 1	93.64	94.84	96.38	94.63
	−1 to 1	94.34	95.08	97.14	96.31

Table 2: Results of CHASPPRAU-Net for spine segmentation on VerSe 2020/2019 datasets with normalization and augmentation.

ent approach that leverages U-Net with a double-layered application. At first, it is used within the network to manipulate data and create a denser dataset.

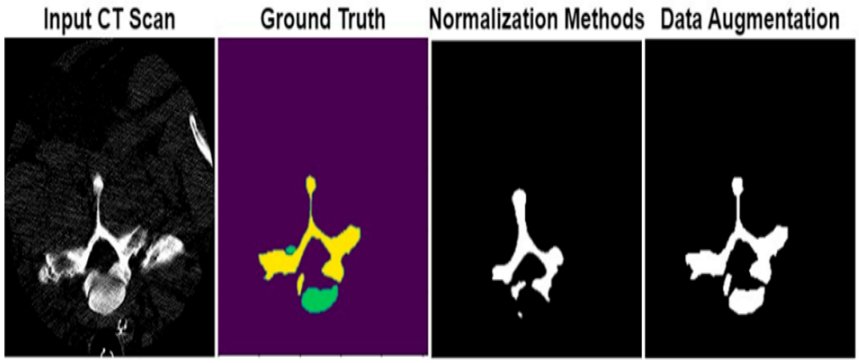


Fig. 6: Image segmentation results of proposed CHASPPRAU-Net model for spine segmentation

This process eventually results in a dataset in which the vertebrae centroids are labeled. The 2D-Dense network is designed by adding residual and dense

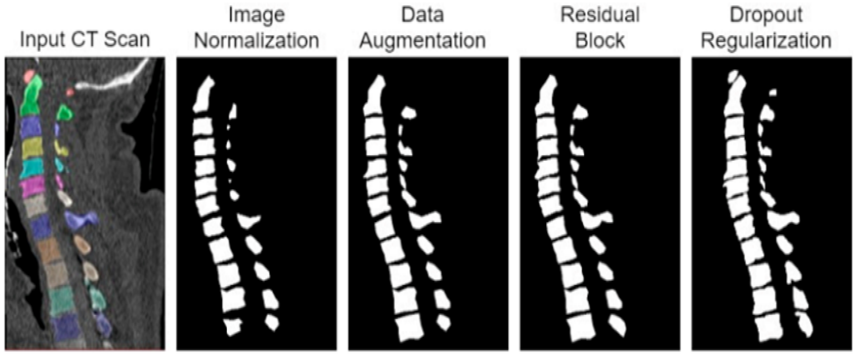


Fig. 7: Image segmentation results for proposed 3D MRU-Net model for spine segmentation

interconnections between the layers. Residual connections are used to transmit information over the whole down or up-sampling blocks, dense interconnections are used to pass unprocessed information to the middle layer of down and up-sampling blocks. This improves accuracy and transferring the finer details the down-sampling would lose in the process otherwise. After applying the network to the dataset, there's a pre-processing stage that includes centroid location estimation based on Aggregation, applying Savitzky-Golay filter and eventually a threshold to eliminate smaller erroneous predictions.

To provide deep segmentation of each vertebra, the Region of Interest (ROI) is determined using the centroid of the identified vertebrae. This ROI is then processed by a modified version of U-Net, specifically the 3D-Dense-U-Net, which

builds upon the original U-Net architecture and 3D-U- Net. Adding dense layers enhances feature propagation and helps in more accurate segmentation.

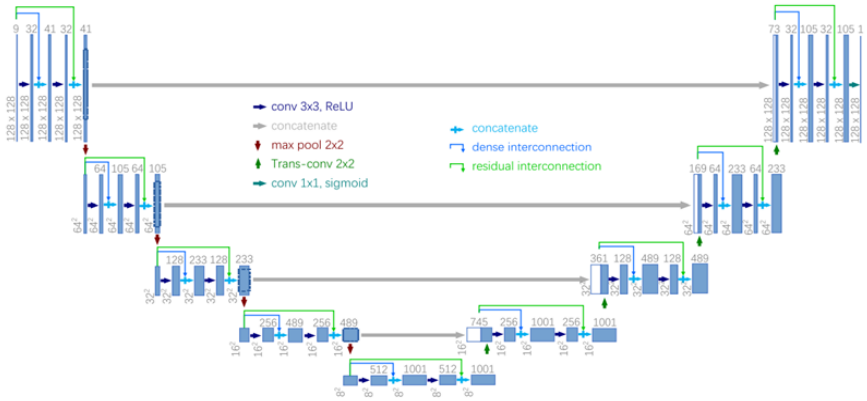


Fig. 8: 2D-Dense-U-Net architecture for vertebrae localization.

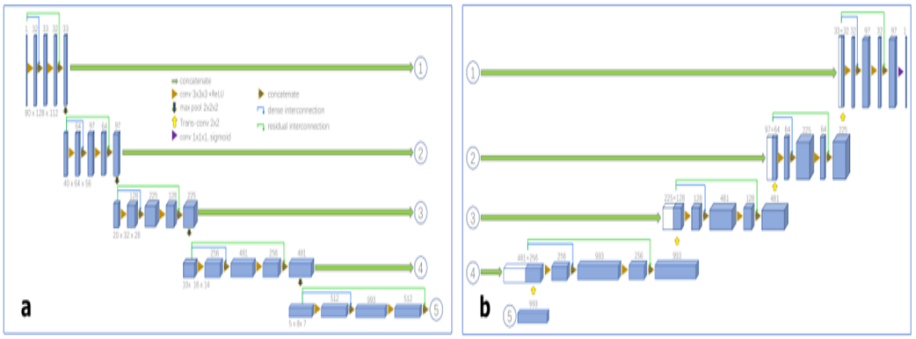


Fig. 9: Proposed 3D-Dense-U-Net architecture

This segmentation method resulted in the following Dice scores on various segmentations:

Metrics	DC	IoU	HD (mm)	PA
Case11	0.951 ± 0.017	0.908 ± 0.031	3.177 ± 1.156	0.998 ± 0.001
Case12	0.955 ± 0.011	0.914 ± 0.019	4.063 ± 1.099	0.997 ± 0.001
Case13	**0.950 ± 0.013**	**0.906 ± 0.023**	4.227 ± 2.637	0.998 ± 0.001
Case14	**0.958 ± 0.010**	**0.919 ± 0.019**	**3.156 ± 1.241**	0.998 ± 0.001
Case15	0.952 ± 0.018	0.909 ± 0.032	**5.443 ± 4.509**	0.997 ± 0.001
All	0.953 ± 0.014	0.911 ± 0.025	4.013 ± 2.128	0.998 ± 0.001

Table 3: Dice scores on various segmentations

Averaging a Dice score of 0.953. For scoliosis, as discussed in Li et al. [11] the standard measurement of the severity of the scoliosis measured by the Cobb angle,Li suggests an automated method to measure the Cobb angle using Deep Learning Framework called "SpineCurve-net" from an input of CT images. Firstly,

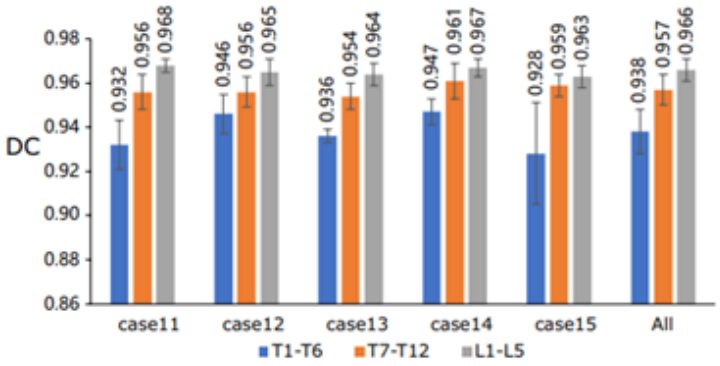


Fig. 10: Evaluation on three groups: upper thoracic, lower thoracic and lumbar spine

the framework used a 3D U-Net which utilizes down sampling to reduce noise and redundancy and to extract more contextual information. The network structure inhibits the correlation between slices such that the spatial and contextual relations between each vertebra is preserved. On the output, applying kernel filtering which makes the model more robust and less vulnerable to biases caused by the data. This process includes applying a circular mask devised through the midpoint circle algorithm and then applying for each voxel a k-means clustering which determines the centroid of the vertebra and ensuring there's enough representation for each vertebra.
Later, three-dimensional NURBS curves were fitted using 3rd degree B-Spline

basis functions. The NURBS-net was designed to predict control points and knot vectors from the spinal segmentation results provided by U-Net. Based on the ResNet architecture, NURBS-net was trained using the same dataset as U-Net. To create more data diversity and improve the model's performance, random affine transformations were applied to both spine segmentation outputs and their corresponding "Ground truth curves" during the training process. Based on the

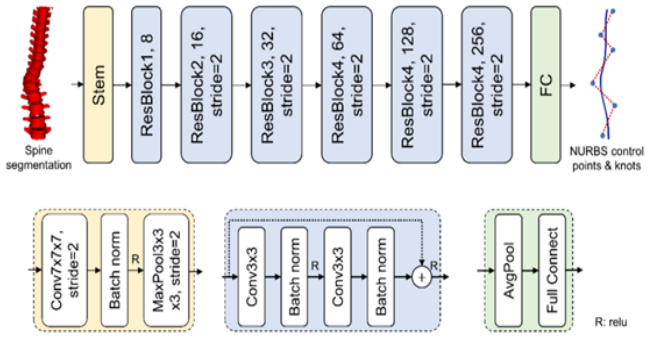


Fig. 11: Res-Net-based deep learning network for spinal curve fitting, using spine segmentation as input and NURBS curves as output.

predicted control points and knot vectors, for each patient, a NURBS-curve was generated with 100 points and corresponding tangent vectors. For each tangent vector, angles between tangent vectors were calculated for the 5th to 95th point (to be less affected by incomplete spine structure modeling caused by imaging at the ends of the scanning). After building a convoluted angle map the projected curve. The maximal angle among all the angles formed by the tangent vectors along the projected curve was calculated. This angle is named "MAP-2D-CA." The accuracy, F1-score, precision, and recall were 0.996 ± 0.002, 0.945 ± 0.026, 0.920 ± 0.047, and 0.973 ± 0.014 respectively in the training dataset, and 0.992 ± 0.062, 0.877 ± 0.062, 0.818 ± 0.099, and 0.954±0.045 in the testing dataset. DC scores weren't published

The use of Spine-transformers and Vertebra labeling via segmentation is detailed in Tao et al. in [21]. The authors suggest that published methods can be broadly divided into two categories: vertebra labeling and vertebra segmentation, when labeling refers to identifying and localizing each vertebra in a given 3D image, without performing segmentation. Although previous studies achieved promising results, they either required additional post-processing to handle arbitrary FOV issues or were difficult to apply to scans with arbitrary FOV due to the use of pre-defined adjacency matrices.

The solution to this problem is Transformers, which are effective at addressing computer vision challenges, making them an ideal choice in this case.

Present Transformers: Despite significant progress achieved the usual trans-

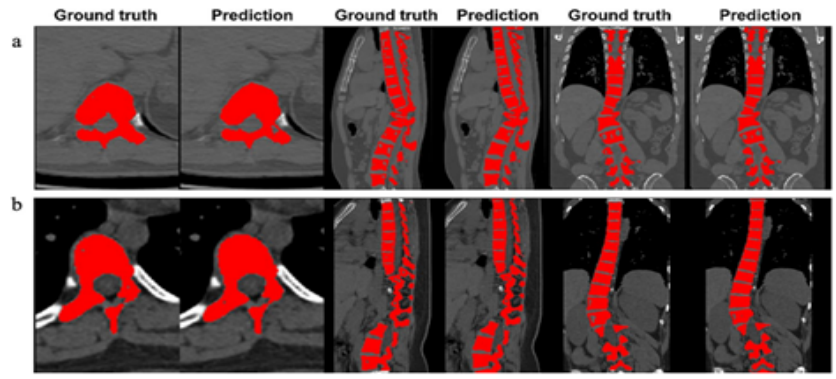


Fig. 12: Spine segmentation results for two patients, with images cropped to remove irrelevant backgrounds, highlighting training and testing dataset outcomes.

former has a problem with object detection from 3D images. The reason is the core of the transformer, The attention module, which identifies complex dependencies between elements of each input data, requires computing similarity scores for all pairs of positions in the input. This scales quadratically with the number of positions, making it too expensive when the number is large.
The architecture of the Spine- Transformers, as presented in Fig.13, The network consists of a CNN backbone, a lightweight transformer with a skip connection and learnable positional embeddings for the encoder and decoder, respectively, and two feed-forward branches that predict the existence of query vertebrae and regress their coordinates. The design includes a skip connection from the backbone CNN to the output of transformer for improved performance. Vertebra detection: The input to the Spine Transformers is fixed-size patches. Spine-Transformers infer all N vertebrae in parallel from an input patch, where NNN is the maximum number of vertebral levels in the ground truth. Each vertebra's ground truth label consists of a binary tag indicating its presence in the patch and, if present, a geometrical description with its center location in 3D and its radius. Classic box detectors are inadequate for vertebra detection because they are not rotational invariant and their performance is sensitive to object orientation. They propose a new rotational invariant detector called InSphere detector to address orientation variations.
Backbone network: a modified ResNet50 architecture is used as the backbone network to extract high-level vertebral features, removing the max pooling layer to maintain higher spatial resolution in the feature maps. Transformer encoder: The bottom-level feature map is passed through a $1\times1\times1$ convolution, reducing the channel dimension and creating a hidden dimension for learnable positional embeddings. This processed feature map is then collapsed and serves as the input to the transformers. Since the attention module in transformers is permutation-invariant, a learnable positional embedding is added to provide spatial context. Transformer decoder: using vanilla transformer decoder design, the transformer

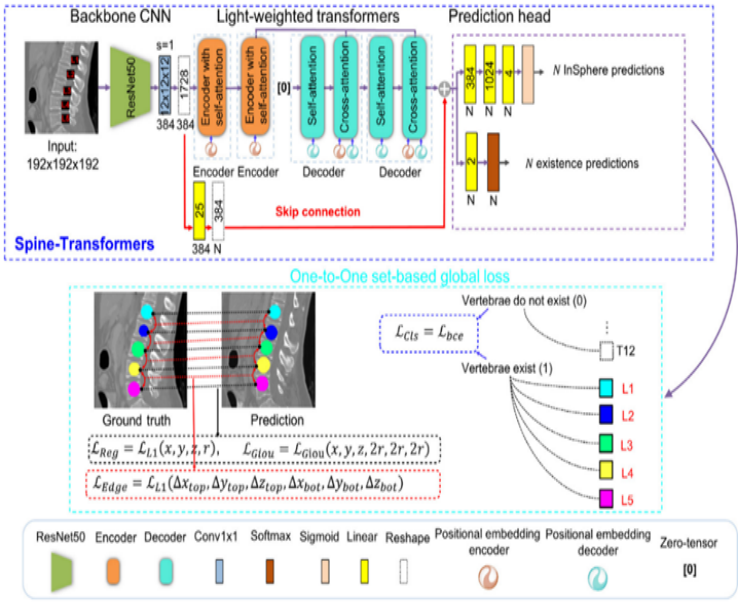


Fig. 13: Spine transformers suggested architecture

decoder yields N predictions of vertebrae in a single forward pass. Like the encoder, a learnable decoder positional embedding is also added to each layer of the decoder.

Skip connection: To reduce typical transformer converge time, a single skip connection is designed from the backbone CNN to the output of the transformer. This structure helps in passing InSphere both, context and gradient information, during the forward and backward phases of training. Light-weighted architecture design: With only two layers of encoders and two layers of decoders the Spine-Transformers feature a light-weighted design. Along with a skip connection, this design strikes a good balance between performance and memory consumption. Joint segmentation and center refinement: Spine-Transformers are

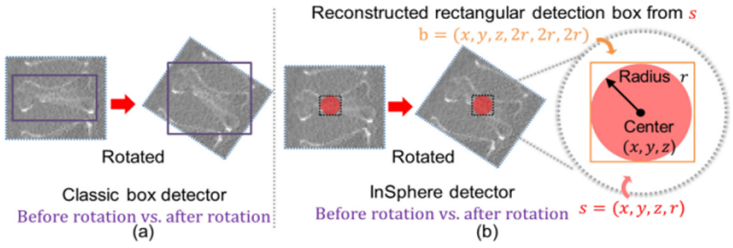


Fig. 14: A comparison of box detector (a) with InSphere detector (b). InSphere detector is not sensitive to vertebral orientation.

trained end-to-end to predict and segment vertebrae in spine CT scans using overlapping patches and a sliding- window method. The tasks of center refinement and segmentation are combined in a single multi-task encoder- decoder network, improving efficiency and accuracy without needing to identify individual vertebrae.
Training: During training, a contextual heat map is generated around the predicted vertebra center, concatenated to the original image and a sub-volume is cropped around the detected center and used as epochs. The model achieved the

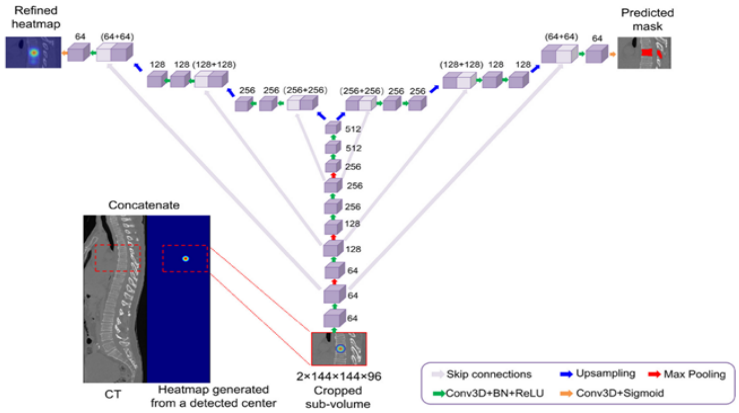


Fig. 15: Architecture of the multi-task encoder-decoder network for joint segmentation and heatmap regression.

following results as measured against the top 3 teams performed in the VerSe 2019 challenge on public and hidden datasets

Evaluated on the public-test data				
Team / Ref. Author	Labeling results		Segmentation results	
	Id-Rate(%)	L-Error (mm)	DOC	HD (mm)
christian_payer / Payer C.	95.65 (100.00)	**4.27** (3.29)	0.909 (0.955)	6.35 (4.62)
iflytek / Chen M.	96.94 (100.00)	4.43 (3.70)	**0.930** (0.960)	6.39 (4.88)
diag / Lessmann N.	89.86 (100.00)	14.12 (13.86)	0.851 (0.943)	8.58 (4.62)
Our method	**97.22** (100.00)	4.33 (4.16)	0.911 (0.950)	**6.34** (4.12)
Evaluated on the hidden-test data				
Team / Ref. Author	Labeling results		Segmentation results	
	Id-Rate(%)	L-Error (mm)	DOC	HD (mm)
christian_payer / Payer C.	94.25 (100.0)	**4.80** (3.37)	0.898 (0.955)	7.08 (4.45)
iflytek / Chen M.	86.73 (100.0)	7.13 (3.81)	0.826 (0.965)	9.98 (5.71)
diag / Lessmann N.	90.42 (100.0)	7.04 (5.30)	0.858 (0.939)	8.20 (5.38)
Our method	**96.74** (100.0)	5.31 (3.78)	**0.901** (0.939)	**6.68** (4.12)

Table 4: Evaluation results table of spine transformers versus different models

3 Background

3.1 YOLOv8

YOLO, by Joseph Redmon et al., was published at CVPR 2016[15]. It presented a real-time end-to-end approach to object detection analysis. Ultralytics, the developers of YOLOv8 as well as YOLOv5, have created a robust computer vision model[6][16]. Unlike earlier versions, YOLOv8 uses an improved structure, providing more flexible bounding boxes, and keeps high level of accuracy with speed, making it versatile for many use cases[6].
Figure.16 presents a detailed description of the YOLOv8 architecture. YOLOv8 uses a special backbone called C2f module which is an improvement to the older CSPLayer.
The C2f module (cross-stage partial bottleneck with two convolutions) combines high-level features with contextual information to improve detection accuracy[15]. YOLOv8 uses an anchor-free design, predicts directly the center of an object, removing the need for predefined anchor boxes and simplifying the model. A decoupled head is used for classification and regression tasks independently, allowing each branch to specialize and improving overall accuracy. In the output layer, a sigmoid function is used as the activation function for object detection score, representing the probability that a bounding box contains an object. SoftMax function calculates the object's probabilities belonging to each possible class[15]. The YOLOv8 architecture features a modified CSPDarknet53 of the

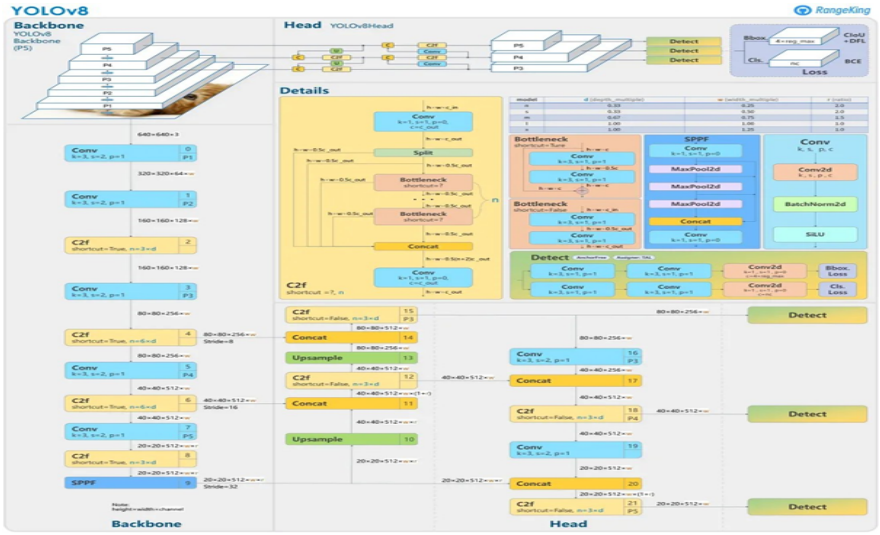


Fig. 16: YOLOv8 Architecture [2]

YOLOv5 backbone with a C2f module, a SPPF layer for faster computation by

pooling features into a fixed-size map, convolution layers with batch normalization and SiLU activation. Its decoupled head processes objectness, classification, and regression tasks separately for improved accuracy.
YOLOv8, which builds on the foundations of YOLOv5, presented several advancements compared to its earlier versions. Evaluated on the MS COCO dataset test-dev 2017, YOLOv8x achieved an AP of 53.9% with an image size of 640 pixels (compared to 50.7% of YOLOv5 on the same input size) with a speed of 280 FPS on an NVIDIA A100 and TensorRT.

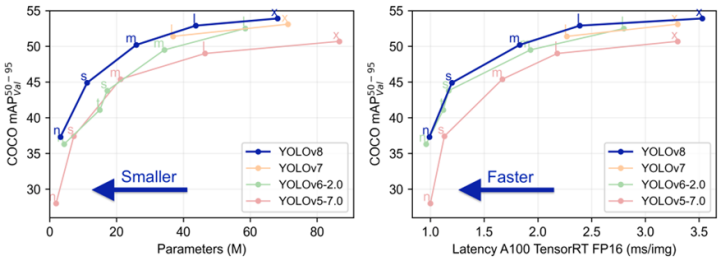


Fig. 17: Latency and parameters convergence comparison through different YOLO versions [6]

3.2 Attention U-Net

Attention U-Net, by Oktay et al, is a CNN with some modification to the known U-Net[12], created to enhance pancreas segmentation, through focusing on relevant areas within a given CT scan. As implies ,the network relies on the Decoder-Encoder U-Net architecture as a base structure, with additional attention gates (AG) between Decoder layers. The network contains down-sampling using max-pool for feature extraction in the encoder part and up sampling layers in the decoder. As the decoder part starts (first layer of up sampling), positioned along the skip connections are the attention gates.
The base structure – U-Net comprised of convolutional, max pool, up-convolutional layers and copy and skip connections. The network applies repeated activation of 2 3X3 unpadded convolutions followed by ReLU. In the Encoder part , the convolutions are then followed by applying 2X2 max-pooling for down-sampling. In the Decoder part ,after applying these conv layers, the network applies up sampling. To minimize contextual information loss, the up-sampled feature maps are combined with the corresponding feature maps from the encoder at the same level (prior to down sampling) to prevent significant contextual loss from the feature map. Initial work of Attention Gates has explored attention-maps by interpreting gradients of output class scores with respect to the input image through natural image analysis, knowledge graphs, and language processing

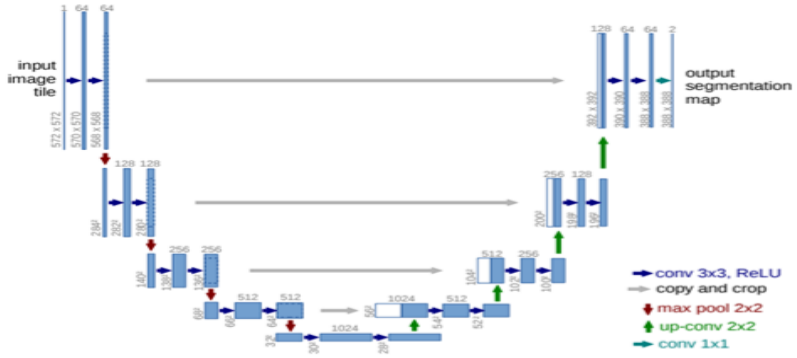


Fig. 18: Original U-Net Architecture [17]

tasks. Through extensive work done in those areas, we can distinguish between two attention mechanisms– Hard and Soft, through their training method.
Hard attention is non-differentiable, which can sometimes require reinforcement learning for parameter updates, while soft attention is differentiable and uses backpropagation for training.
The attention modules used in this model are soft attention based and designed to improve accuracy, by reducing false positive predictions through suppression of background features in background regions, without scaling down the region of interest.
Attention coefficients: $\alpha_i \in [0, 1]$, identify noticeable image regions and remove feature responses to keep only the relevant regions to the specific task. The output of the Attention gate is element-wise multiplication of the input feature maps and Attention coefficients: $\hat{x}^l_{i,c} = x^l_{i,c} \cdot \alpha^l_i$. A single scalar attention value is computed for each pixel: $x^l_i \in \mathbb{R}^{F_l}$, in case of usage in multiple semantics, an attention vector for each pixel is calculated. That causes for each one of the AGs to focus on different targeted section within the feature map. Gating vector $g^l_i \in \mathbb{R}^{F_g}$ is used for each pixel I to determine focus regions. The gating vector contains informational context used to prune less relevant feature responses. To achieve high accuracy model there's a usage in additive attention and formulated as follows:

$$q^l_{\text{att}} = \psi^T \left(\sigma_1 \left(W_x^T x^l_i + W_g^T g_i + b_g\right)\right) + b_\psi, \tag{1}$$

$$\alpha^l_i = \sigma_2 \left(q^l_{\text{att}} \left(x^l_i, g_i; \Theta_{\text{att}}\right)\right), \tag{2}$$

where σ_2 corresponds to Sigmoid activation function. AG is characterized by a set of parameters Θ_{att} containing linear transformations $W_x \in \mathbb{R}^{F_l \times F_{int}}, W_g \in \mathbb{R}^{F_l \times F_{int}}, \psi \in \mathbb{R}^{F_l \times 1}$ and bias terms $b_\psi \in \mathbb{R}, b_g \in \mathbb{R}^{F_{int}}$. The linear transformations are computed using channel-wise $1 \times 1 \times 1$ convolutions for the input tensor. In the Attention U-Net, the Attention gates are incorporated through usage of skip connections and the up-sampling stage in each level of the decoder. Each

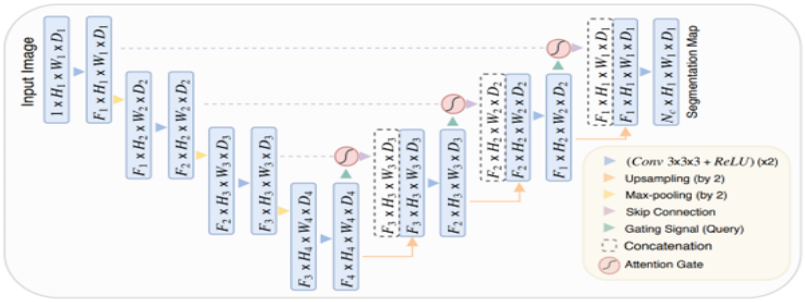


Fig. 19: Attention U-Net model architecture

attention block receives two inputs: The gating signal which is the feature information being received from the previous stage in the decoder, and the spatial information that is received from the encoder through the skip connection.
On both inputs a convolution, is applied: is with (1,1) stride while is with stride of (2,2) which causes both inputs to be in the same size. After that, we can sum the weights, that causes aligned weights get larger while unaligned weights get relatively smaller and pass the weights through a Relu activation function. Later, function is applied resulting a matrix size of $L \times W$.
The received weights matrix is needed to be normalized and to apply multiplication with X, so Sigmoid function and later an Up-sample is applied on the output. The result from the module is vector based on relevance and handled later as usual in the normal U- Net flow.

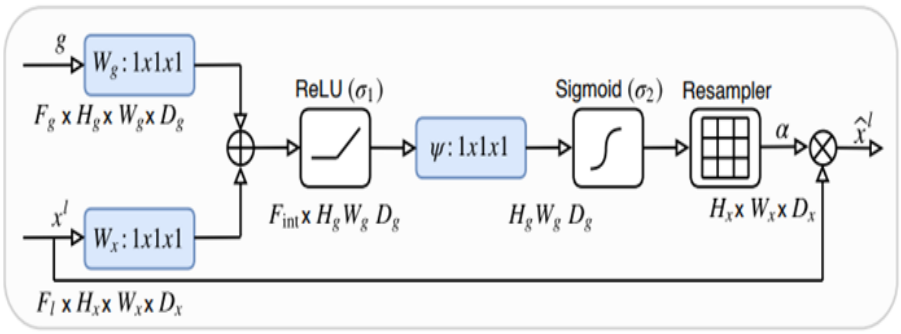


Fig. 20: Attention gate scheme within the Attention U-Net

3.3 B-Spline

B-Splines are one of the most promising curves in computer graphics and nowadays They are a fundamental tool in computer aided design industry (CAD) due

to their superior geometric properties.
B-Splines are a generalization of Bezier curves, constructed using an orthonormal basis of recursive functions. They are composed of polynomial curve sections, with a degree matching the B-Spline's degree, joined at knots where parametric function discontinuities occur. The shape of the curve is influenced by control points, and it is represented parametrically as a weighted sum of basis functions and control points [10].
The degree of the basis function influences the curve: higher degrees provide more flexibility and smoother transitions but require longer computation times. Considering a,b,c coordinates with respect to parameter t, B-Spline can be represented as: a = a(t),b = b(t),c = c(t). Considering N control points the polygon is defined by connecting the points with linear lines. The control polygon provides a visual guide to the curve's structure, Although the curve typically does not pass through the control points, the polygon helps in understanding and editing the curve. There are N-n section when n is the polynomial degree, each one of these section is joined by N-n-1 knots. For control point cp=<x,y,z> [10]. B-Spline's parametric equation is defined as:

$$V(t) = \sum_{k=1}^{N} BS_{k-1}^{n}(t)\,\mathrm{cp}_k, \quad t_n \le t \le t_N, \quad N \ge n+1$$

The knots are tied by the parameter t. The t_j represent the transitions between the j and $j+1$ polynomial segments [10]. Choosing the number of knots can be challenging: too many knots may lead to data overfitting, while too few may cause underfitting [7]. The values t_j are monotonically increasing and may be equally spaced, integers, or positive values [10]. The functions $BS_k^n(t)$ are defined recursively as:

$$BS_k^n(t) = \frac{t - t_k}{t_{k+n} - t_k} BS_k^{n-1}(t) + \frac{t_{k+n+1} - t}{t_{k+n+1} - t_{k+1}} BS_{k+1}^{n-1}(t) \tag{2}$$

With the unit step function being defined as:

$$us(ts) = \begin{cases} 1, & t > 0, \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

The 0^{th} order polynomial BS is:

$$BS_k^0(t) = us(t - t_k)\, us(t_{k+1} - t) \tag{4}$$

The left graph present the quadratic B-spline basis functions. We can see the overleap and sum up to 1, where the spline is defined. The B- spline has 8 control points when the first and last points are not connected to the spline because the function is not defined at those points (it can be fixed using open uniform B-Spline)[19].

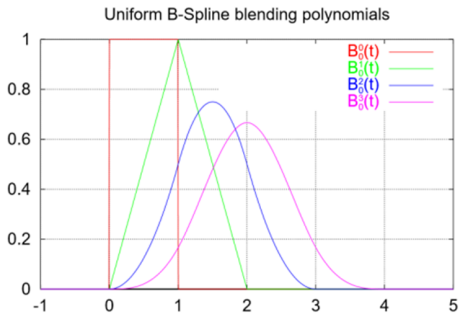
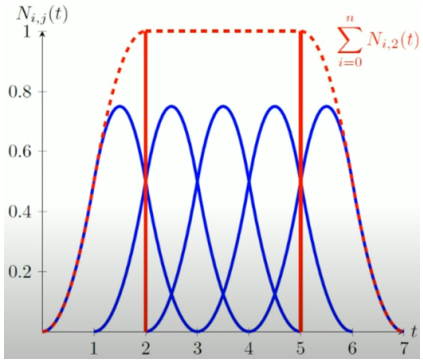
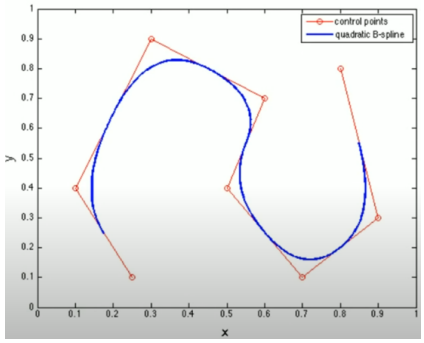


Fig. 21: Four lowest order blending polynomials corresponding to uniform B-Splines



(a) Basis functions and summation curve.



(b) Quadratic B-spline curve.

Fig. 22: Visualization of B-spline basis functions and the resulting curve.

4 Proposed Approach

The purpose of this stage is to use a CT scan as an input for a YOLOv8 algorithm. YOLOv8 creates a bounding box around each vertebra in a given spine scan, thereby delineating each vertebra. The segmentation will be improved using an Attention U-Net, utilizing its ability to understand contextual information, to enhance our ability to segment each vertebra.
Each segmented vertebra will used to create points in space $(x_0, y_0), \ldots, (x_n, y_n)$. These control points are subsequently used to construct a cubic B-Spline curve (k = 3). The B-Spline is defined using a non-uniform knot vector, allowing for greater flexibility and local control over the spline's shape. This ensures that the curve accurately represents the spatial alignment of the vertebrae.
The spine data will be collected from given scans and aggregated to an average, which will serve as the basis for determining the ground truth. Once the spine has been segmented and a B-Spline has been generated, we compute the

differences using the distance function defined as follows:

$$d(C_{truth}, C_{given}) = \|C_{truth}(u) - C_{given}(v)\|$$

where $C_1(u)$ and $C_2(v)$ represent the positions on the respective curves at parameters u and v, and $\|\cdot\|$ denotes the Euclidean norm. By calculating the dis-

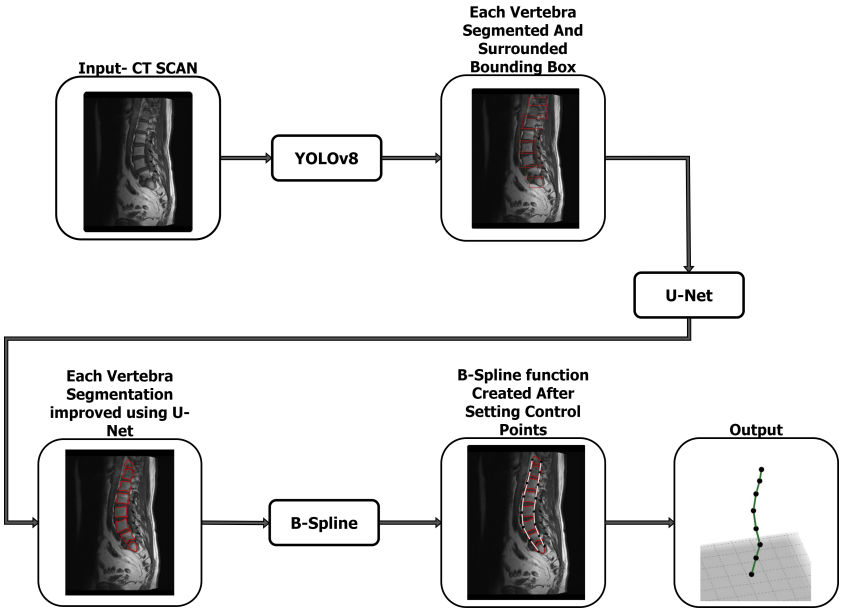


Fig. 23: BAYU-Net Architecture

tance from the ground truth through sampling multiple points scattered on the B-Spline we can estimate where and how severe the differences are and locate the problematic area. The model we have proposed uses **B**-Splines **A**ttention **Y**OLOv8 and **U-Net** to solve the problem of scoliosis analysis.

5 Dataset

We will use RSNA 2024 Lumbar Spine Degenerative Classification dataset, (https://www.rsna.org/rsnai/ai-image-challenge/lumbar-spine-degenerative-classification-ai-challenge) the dataset contains around 147,000 DICOM images which are the standard format for medical imaging.

6 Research Process

The most important part of our process has been maintaining open communication, being honest about our knowledge, and balancing teamwork with independent work. At the project's start, we set clear expectations and established

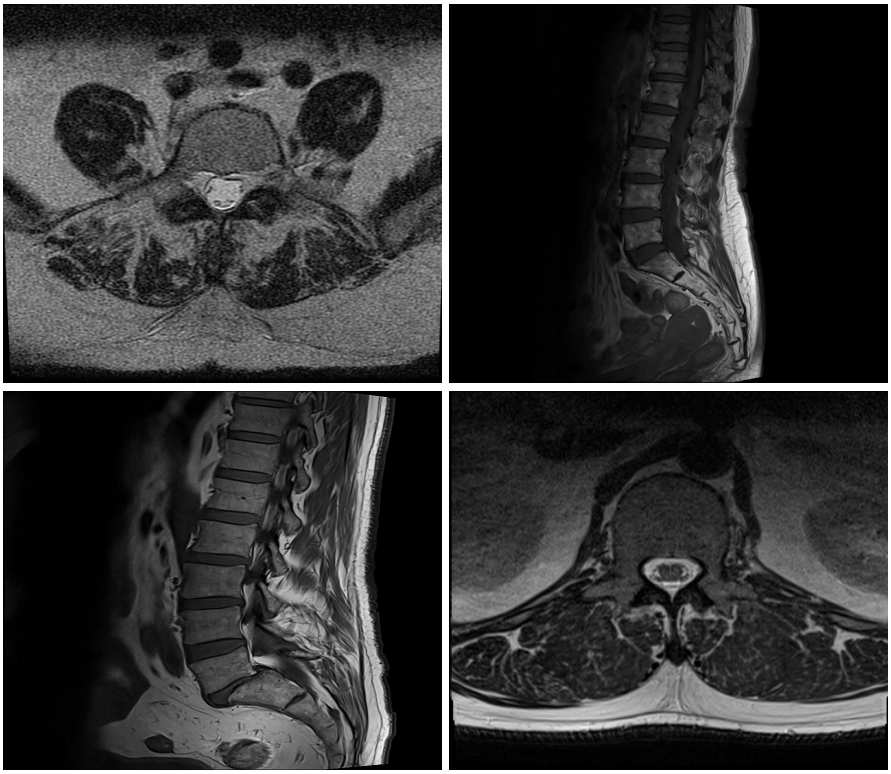


Fig. 24: Samples from RSNA 2024 Lumbar Spine founf in https://www.rsna.org/rsnai/ai-image-challenge/lumbar-spine-degenerative-classification-ai-challenge

ground rules to guide our collaboration.
Once we fully understood the problem, we focused on researching the spine, scoliosis, and medical terminology to build a solid foundation. At the same time, we explored advanced machine-learning techniques.

6.1 Hyper-parameter Optimization

To create the most accurate B-Spline representation for the given scan, we need to provide the best segmentation results to the B-Spline modeling. Therfore, we need to explore different hyperparameters and determine the best combination to optimize the segmentation using YOLOv8 and U-Net Attention. The hyperparameters for the YOLO section in our proposed architecture:

- **Learning Rate:** $1 \cdot 10^{-3}$, $1 \cdot 10^{-4}$, $1 \cdot 10^{-5}$
- **Batch Size:** 8, 16, 32

- **Confidence Threshold:** 0.25, 0.5
- **IoU Threshold:** 0.5, 0.7
- **Epochs:** 50, 100
- **Augmentations:** Flip, Rotation
- **Anchor Auto-tuning**: Enabled, Disabled

The next phase is to explore hyperparameters and fine-tune the Attention U-Net section for more improved segmentation. The hyperparameters for this section are:

- **Learning Rate:** $1 \cdot 10^{-3}$, $1 \cdot 10^{-4}$, $1 \cdot 10^{-5}$
- **Epochs:** 50,100,150
- **Batch Size**: 4, 8, 16
- **Dropout Rate**: 0.2, 0.5
- **Loss Function**: Dice Loss, Combined Dice + BCE Loss

. In the implementation phase, we will need to use Google Colab

7 Expected Achievements

- **Accurate Vertebra Segmentation:** Achieve higher-accuracy segmentation of individual vertebrae from the existing solution out today.
- **Precise Scoliosis Curve Modeling:** Develop a cubic B-Spline model that accurately shows the spine's shape and curvature, helping to analyze scoliosis more precisely.
- **Research Contributions and Clinical Application:** Develop a system that is both efficient and reliable for clinical applications and advances research in medical image analysis.

8 Challenges

Firstly, we almost didn't encounter any research paper in our field, during our studies before the project besides the course "Data mining and machine learning" which gave us limited understanding about this world. We developed the ability and experience to read research papers, especially research on medical and complicated machine learning terms.
Secondly, Our project includes YOLOv8 and Attention U-Net architectures. To understand these architectures fully we had to research from the ground up - from the original U-net and YOLO papers to the most up-to-date versions.
By exploring related work we were able to understand existing methods to segment the spine and identified ways to improve them for scoliosis analysis through our proposed architecture.

9 GUI

The GUI allows users to upload medical imaging scans, such as CT images, directly into the system for processing. It provides an intuitive interface for visualizing both the original scans and the segmented vertebrae, with options to zoom in, adjust contrast, and overlay segmentation results. Additionally, users can export the segmentation data and scoliosis curve models for further analysis or incorporation into clinical reports. Table 5 depicts the GUI initial posibilities.

GUI First screen

Loading scan screen

Scoliosis detected through comparison with ground truth

Fig. 25: Graphical User Interface (GUI) overview.

ID	Test Description	Expected Result
001	Click "Upload" button	Opens file dialog
002	Upload a valid image (JPEG, PNG)	Image is displayed in the GUI
003	Upload an invalid file type (PDF)	Error message: "Unsupported file type"
004	Upload a large image file	Image is resized and displayed correctly in the GUI
005	Cancel file upload	No image is displayed; GUI remains unchanged

Table 5: Test Cases for GUI.

10 Testing and Verification Plan

ID	Requirement Description	FR/NFR
1.0	The system shall allow uploading an image.	FR
1.1	The image shall be uploaded by the user.	NFR
1.2	The image shall be formatted as PNG/JPEG.	NFR
2.0	The system shall preform segment the photo.	FR
2.1	The segmentation will be preformed using BAYU-Net architecture.23	NFR

Table 6: Requirements Table

References

1. Anatomy, T.M.: Bones of the back (2025), https://teachmeanatomy.info/back/bones/
2. Blog, R.: What is yolov8? (2025), https://blog.roboflow.com/what-is-yolov8/
3. Cheng, P., Yang, Y., Yu, H., He, Y.: Automatic vertebrae localization and segmentation in ct with a two-stage dense-u-net. Scientific Reports **11**(1), 22156 (Nov 2021). https://doi.org/10.1038/s41598-021-01296-1, https://doi.org/10.1038/s41598-021-01296-1
4. clevelandclinic: Spine structure and function (2023), https://my.clevelandclinic.org/health/body/10040-spine-structure-and-function
5. clevelandclinic: Scoliosis (2024), https://my.clevelandclinic.org/health/diseases/15837-scoliosis
6. Documentation, U.: Yolov8 (2023), https://docs.ultralytics.com
7. Eilers, P.H.C.: Flexible smoothing with b-splines and penalties (1996), https://projecteuclid.org/journals/statistical-science/volume-11/issue-2/Flexible-smoothing-with-B-splines-and-penalties/10.1214/ss/1038425655.full
8. Fritz Hefti, M.D., P.: Pediatric orthopedics in practice - cobb and scoliosis (2007), https://books.google.co.il/books?id=VRnFkfvRT4EC&pg=PA56&source=gbs_selected_pages&cad=1#v=onepage&q=cobb&f=false

9. Frost, B.A., Camarero-Espinosa, S., Foster, E.J.: Materials for the spine: Anatomy, problems, and solutions. Materials **12**(2) (2019). https://doi.org/10.3390/ma12020253, https://www.mdpi.com/1996-1944/12/2/253
10. Gordon, W.J.: B-spline curves and surfaces, https://doi.org/10.1016/B978-0-12-079050-0.50011-4
11. Li, L., Zhang, T., Lin, F., Li, Y., Wong, M.S.: Automated 3d cobb angle measurement using u-net in ct images of preoperative scoliosis patients. Journal of Imaging Informatics in Medicine (Aug 2024). https://doi.org/10.1007/s10278-024-01211-w, https://doi.org/10.1007/s10278-024-01211-w
12. Oktay, O.: Attention u-net: Learning where to look for the pancreas (2024), https://openreview.net/pdf?id=Skft7cijM
13. Orthopedics, Traumatology: Idiopathic scoliosis (2023), https://www.elitenicosia.com/en/cocuklarda-ve-ergenlerde-idiyopatik-skolyoz/
14. Radiopaedia: Cobb angle (2024), https://radiopaedia.org/articles/cobb-angle?lang=us
15. Redmon, J.e.a.: You only look once (yolo) (2025), https://arxiv.org/pdf/1506.02640
16. Reis, D.: Real-time flying object detection with yolov8 (2024), https://arxiv.org/pdf/2305.09972
17. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation (2015), https://arxiv.org/abs/1505.04597
18. Saeed, M.U., Dikaios, N., Dastgir, A., Ali, G., Hamid, M., Hajjej, F.: An automated deep learning approach for spine segmentation and vertebrae recognition using computed tomography images. Diagnostics **13**(16), 2658 (2023)
19. Siach, D.J.: Mathematicsofcomputergraphicsandvirtualenvironments8 (2015), https://www.youtube.com/watch?v=qhQrRCJ-mVg&ab_channel=MathematicsofComputerGraphicsandVirtualEnvironments
20. Stokes, I.A.F.C.: Three-dimensional terminology of spinal deformity (1994), https://journals.lww.com/spinejournal/abstract/1994/01001/three_dimensional_terminology_of_spinal_deformity_.20.aspx
21. Tao, R., Liu, W., Zheng, G.: Spine-transformers: Vertebra labeling and segmentation in arbitrary field-of-view spine cts via 3d transformers. Medical Image Analysis **75**, 102258 (2022). https://doi.org/https://doi.org/10.1016/j.media.2021.102258, https://www.sciencedirect.com/science/article/pii/S1361841521003030
22. Unknown, A.: Congenital scoliosis (2003), https://link.springer.com/article/10.1007/s00586-003-0555-6
23. Unknown, A.: Adolescent idiopathic scoliosis: What you need to know (2021), https://skoliosis.my/scoliosis-types/adolescent-idiopathic-scoliosis/