



Scoliosis Detection: Edge-Preserving Preprocessing of Spinal X-Rays Using PDEs and Deep Learning-Based Classification

**Systems & Biomedical Engineering Department
Special Functions & Partial Differential Equations
(MTH2245)
Chasing Arcs**

ARC'S TEAM

N ame	B.N.	ID
Ekram Ahmad Auf Mohamed	12	9230228
Raghad Abdelhameed Abdelhady Abdelhameed	28	9230371
Zeyad Ashraf Ahmed Mohamed	30	9230391
Salma Ali Ibrahim Ali	35	9230439
Abdlrhman Reda Khalaf Mohamed	42	9220428

Table of Contents

1.	Abstract	5
2.	Introduction	6
3.	Problem Definition	8
	3.1 Problem	8
	3.2 Objective	8
	3.3 Mathematical Formulation	8
	3.3.1 Pre-processing Stage	8
	3.3.2 Classification Stage	9
	3.3.3 Line Segmentation	9
4.	Dataset	10
5.	Methodology	11
	5.1 Heat Equation	11
	5.2 Anisotropic Diffusion	13
	5.3 Classification Stage using CNN	14
	5.4 U-Net for Binary Segmentation	19
	5.5 Cobb Angle Calculation Using Automated Image Analysis	19
6.	Experimental Work	21
	6.1 Data Description	21
	6.2 Classification Model Architecture	21
	6.3 Training Methodology	23
	6.4 Line Segmentation Using U-Net	23
7.	Results	25
	7.1 Comparison of Heat Equation and Anisotropic Diffusion for Pre-processing Spinal X-ray Images	25
	7.2 Structural Similarity Index Measurements (SSIM)	27
	7.3 Results of the CNN model's performance	28
	7.4 Results of Line Segmentation Using U-Net	30
	7.5. Cobb Angle Analysis and Results	31
8.	Conclusion	32
9.	References	33
10.	Appendix	35

List of Abbreviations

• AIS	Adolescent Idiopathic Scoliosis
• PDEs	Partial Differential Equations
• CNN	Convolutional Neural Network
• 3D	Three-Dimensional
• HDE	Heat Diffusion Equation
• PM	Perona and Malik
• PhD	Doctor of Philosophy
• PT	Physical Therapist
• MSc	Master of Science
• API	Application Programming Interface
• ReLU	Rectified Linear Unit
• 1D	One-Dimensional
• SSIM	Structural Similarity Index Measurements
• U-Net	U-shaped Network
• SGD	Stochastic Gradient Descent
• Adam	Adaptive Moment Estimation

1. Abstract

Scoliosis is an abnormal curvature of the human spinal column, often accompanied by vertebral rotation. Adolescent Idiopathic Scoliosis (AIS) is the most common type, typically affecting children between 8 and 18 years old during bone growth at its maximum rate. Early detection of scoliosis often allows for non-surgical treatment options, such as physiotherapy and using specific braces like the Boston Brace or Milwaukee Brace. This research proposes an innovative approach to enhance the accuracy of spinal X-ray image analysis. We aim to improve image quality and diagnostic accuracy by combining the power of Partial Differential Equations (PDEs), specifically the Heat Equation and Anisotropic Diffusion, for pre-processing and a Convolutional Neural Network (CNN) for classification. The pre-processing stage reduces noise and preserves fine details, while CNN classifies images as normal or indicative of scoliosis. Furthermore, we incorporate line segmentation using U-Net to assist in accurate Cobb angle measurements, an essential aspect of scoliosis assessment. We expect significant improvements in noise reduction, edge preservation, and Cobb angle measurement accuracy, leading to higher classification precision and aiding in early diagnosis and efficient treatment planning for scoliosis.

2. Introduction

Scoliosis, a 3D deformation of the human spinal column, is characterized by a lateral curvature of the spine and axial rotation of the vertebrae. Adolescent Idiopathic Scoliosis (AIS) is the most common type, comprising about 80% of pediatric scoliosis and affecting about 3% of adolescents under the age of 16. Scoliosis treatment is highly dependent on the shape and degree of spinal curvature, and while guidelines have been established for best management practices, individual treatment options are based on surgeon experience. Therefore, the development of a clinically validated, patient-specific model of the spine to aid surgeons in understanding AIS in its early stages and inform optimal surgical and non-surgical treatment could provide substantial clinical value [1].

Medical imaging systems are important in physicians' diagnosis processes [2,3]. Using medical images, physicians have information about the internal parts of the body and the organs [4]. Doctors can diagnose complex diseases much more accurately Through medical images that are different from other image types in terms of the information obtained by pixels [5]. Different types of imaging techniques are used for the different parts of the human body [6].

X-ray imaging is a technique that is mostly used in the regions between the bones and the tissue around the bones [7]. X-rays are a valuable diagnostic tool that uses electromagnetic radiation to create images of the inside of the body, aiding in the detection of various medical conditions such as fractures, tumors, and infections. They are non-invasive, relatively quick, and widely available, making them an essential part of modern healthcare. However, one of the disadvantages of X-ray technology is the potential for noise in the images, which can sometimes obscure details and make diagnosis more difficult [8]. Noise in medical X-ray images is primarily categorized into quantum mottle, which is related to the number of incident X-rays, and artificial noise is due to the grid, etc., [9]. Noise is usually generated in the process of acquisition and transmission of images; this affects the visual quality of an image.

Denoising the image without affecting the edge information is a challenging problem in image processing [10-14]. Denoising is usually used as a pre-processing step in many image applications such as medical applications, image segmentation, edge detection, image enhancement, and image decomposition. Filter-based methods such as Median filter, and Gaussian filter [15,16] are commonly used for this purpose, as these are simple and easy to implement. The main drawback of these filters is, these filters annihilate important details and blur prominent geometrical edges because a fixed filter is applied to all pixels of the image, whether these pixels are noisy or not. Therefore, the median and Gaussian filters are not appropriate choices for many image-smoothing applications [17-19].

Techniques based on partial differential equations (PDEs) are widely used in image denoising [20]. According to [21] In 1984, the heat diffusion equation (HDE) in image processing to obtain a parametric family generated by an image. HDE is an isotropic heat diffusion equation with uniform diffusion of the image in all directions that causes the edges to blur during smoothing [22].

Results obtained from isotropic diffusion are similar to the results obtained from Gaussian convolution of the image [23, 24]. To overcome this drawback of isotropic HDE, Perona and Malik (PM) proposed anisotropic diffusion [25]. Perona and Malik, for the first time, used Anisotropic diffusion for image smoothing without affecting the edge information much by decreasing diffusivity near edge locations [27]. In the PM method, the diffusion coefficient function is taken as a decreasing function of the gradient magnitude of the image varying with space and time [28-30].

Various techniques via anisotropic diffusion have been developed to remove noise and edge preservation [31] simultaneously. These techniques are based on gradient vector flow [32-35]. A fast and flexible denoising convolutional neural network (CNN) has been applied for image denoising [36]. This method works on the down-sampled sub-images to give effective results. This convolutional neural network (CNN) based technique performs better for synthetic noisy images [37,38]; however, their performance is limited to real noisy images [39-42]. The CNN model was trained on a dataset of spine X-ray images, with each image labeled as either "scoliosis" or "normal." The model successfully learned to identify key features associated with scoliosis, such as spinal curvature and vertebral rotation, enabling accurate classification of new, unseen X-ray images. This automated approach has the potential to significantly improve the efficiency and accuracy of scoliosis diagnosis, leading to earlier intervention and better patient outcomes [43-46].

The main objective behind all the above-mentioned approaches is the application of the heat equation and anisotropic diffusion for pre-processing X-ray images. The proposed approach involves a two-step process: initial noise reduction using the heat equation, followed by edge-preserving smoothing using anisotropic diffusion, leading to improved diagnostic accuracy. followed by a Convolutional Neural Network (CNN) for classification. The primary objective is to develop a robust and efficient method for scoliosis detection that overcomes the limitations of existing approaches and achieves superior performance. Additionally, the project aims to perform line segmentation using U-Net of the spine in X-ray images to facilitate accurate Cobb angle measurements, enhancing the model's utility for scoliosis diagnosis.

3. Problem Definition

3.1. Problem:

Scoliosis, a lateral curvature of the spine, is a common condition that can lead to significant health issues if left untreated. Early detection is crucial for effective treatment. However, accurate diagnosis from spinal X-rays can be challenging due to noise and image degradation, which can hinder the performance of automated classification algorithms.

3.2. Objective:

The goal is to improve the accuracy of a Convolutional Neural Network (CNN) model for classifying spinal structures while ensuring that the model retains crucial information about the spinal anatomy. Additionally, the project aims to perform line segmentation using U-Net of the spine in X-ray images to facilitate accurate Cobb angle measurements, enhancing the model's utility for scoliosis diagnosis.

3.3. Mathematical Formulation:

3.3.1. Pre-processing Stage:

1. Heat Equation [50]:

$$\frac{\partial u}{\partial t} = \alpha \nabla^2 u$$

Where:

- $u(x,y)$ is image intensity at position (x, y) and time t .
- $\frac{\partial u}{\partial t}$ is the time derivative of the image intensity.
- α is diffusion coefficient, controlling the rate of diffusion.
- ∇^2 is the Laplacian operator, measuring the curvature of the image intensity.

Apply the heat equation to reduce noise in the image by diffusing the image intensity over time, thereby smoothing out noise.

2. Anisotropic Diffusion [30]:

$$\frac{\partial I_t(x,y)}{\partial t} = \nabla \cdot (c(|\nabla I_t|) \nabla I_t)$$

Where:

- $I_t(x, y)$ is the image intensity at position (x, y) and time t .
- ∇ is the gradient operator.
- $c(|\nabla I_t|)$ is a diffusion coefficient that depends on the gradient magnitude.

Apply anisotropic diffusion to preserve edges while reducing noise, with the diffusion coefficient $c(\nabla I)$ functioning as a gradient magnitude-based factor for selective smoothing.

3.3.2. Classification Stage:

1. Convolutional Neural Network (CNN):

- o Train a CNN model on a large dataset of labeled spinal X-ray images.
- o CNN extracts relevant features from the pre-processed images and classifies them as normal or scoliotic [\[39-42\]](#).

3.3.3. Line Segmentation:

1. U-Net Segmentation [\[53\]](#):

- o The input image $I \in R^{H*W}$ is passed through an **encoder-decoder** architecture with skip connections. The output is a binary segmentation mask $\widehat{M}(x, y)$ obtained via a sigmoid activation:

$$\widehat{M}(x, y) = \sigma(\text{Conv2D}(f_{dec}^0))$$

The model is optimized using **binary cross-entropy loss**.

2. Landmark Detection:

- o Key vertebral landmarks are extracted from the segmentation mask to identify the upper and lower vertebral endplates.

3. Angle Calculation:

- o Tilt angles θ are computed using the slope between vertebral endplate coordinates

4. Cobb Angle Measurement:

- o The severity of scoliosis is quantified using the Cobb angle θ_c . [\[43-46\]](#).

By combining these techniques, the goal is to enhance the quality of spinal X-ray images, making them more suitable for accurate automated classification and, ultimately, improving the early detection and management of scoliosis.

4. Dataset

This research aims to detect scoliosis early, as early diagnosis can prevent the need for surgery. In the early stages, treatment primarily involves bracing and physiotherapy. Common types of braces used are [\[51,52\]](#) :

- **Boston Brace:** This is a rigid brace that applies pressure to correct spinal curvature.
- **Milwaukee Brace:** This brace extends from the chin to the pelvis and is used for more severe curvatures.



Boston Brace



Milwaukee Brace

To achieve this objective, we collected a comprehensive dataset of spinal X-ray scans from various sources, with a specific focus on **Egyptian patient data**. Our data sources include:

- **Clinics specializing in scoliosis treatment:**
 - o ARC for Scoliosis Physiotherapy (Dr. Mahmoud Ibrahim, PhD, PT)
 - o ScolioCare (Dr. Sarah M.Ali, MSc, PT)
 - o Qawam for Spinal Rehabilitation (Dr. Tayseer Saber Abdeldayem)
 - o 4kids Therapy Clinic (Dr. Sarah M.Ali, MSc, PT)
- **Open-source datasets:**
 - o <https://data.mendeley.com/datasets/xkt857dsxk/1> (A dataset of scoliosis, spondylolisthesis, and normal vertebrae X-ray images)
 - o <https://universe.roboflow.com/scoliosis/scoliosis-myka6/dataset/1> (Scoliosis dataset from Roboflow universe)

This diverse dataset, comprising over 580 normal scans and over 765 scoliosis scans, will allow us to train a robust and generalizable model for accurate scoliosis detection in X-ray images.

5. Methodology

5.1. Heat Equation: A Smoothing Technique for Noise Reduction

The heat equation, a partial differential equation, is employed in image processing to model intensity diffusion, where pixel values are treated like temperatures. This diffusion process causes high-intensity (brighter) areas to spread toward lower-intensity (darker) areas, creating a smoothing effect that reduces sharp edges and noise. Applied to medical images, this technique enhances subtle features critical for scoliosis detection. By adjusting parameters, the method optimizes clarity and sensitivity, facilitating early detection by refining image details and minimizing noise.

Mathematically, the heat equation is expressed as [\[50\]](#):

$$\frac{\partial u}{\partial t} = \alpha \nabla^2 u$$

Where:

- $u(x,y)$ is image intensity at position (x, y) and time t .
- $\frac{\partial u}{\partial t}$ is the time derivative of the image intensity.
- α is diffusion coefficient, controlling the rate of diffusion.
- ∇^2 is the Laplacian operator, measuring the curvature of the image intensity.

Implementation Steps:

1. Discrete Approximation of the Laplacian: In a discrete grid (image matrix), the Laplacian at pixel (i, j) can be approximated by the second difference:

$$\nabla^2 u(i, j) \approx u(i+1, j) + u(i-1, j) + u(i, j+1) + u(i, j-1) - 4u(i, j)$$

This approximation sums the values of the neighboring pixels, subtracting four times the center pixel's value, capturing the “smoothness” at that point.

2. Finite Difference Scheme for Time Evolution: To solve the heat equation numerically, we use a finite difference scheme in time as well. Discretizing in time with a small timestep dt , we update the pixel value $u(i, j)$ at each step:

$$u(i, j)^{\text{new}} = u(i, j)^{\text{old}} + \alpha \cdot dt \cdot \nabla^2 u(i, j)$$

In each iteration, this equation updates $u(i, j)$ based on the difference between the center pixel and its neighbors.

3. **Adding Edge Preservation with Diffusivity:** To preserve edges, we adjust the diffusion coefficient dynamically based on the image gradient (sharp changes in pixel values). High gradients indicate edges, so we decrease the diffusivity there to avoid blurring:

- **Compute Gradients:** For pixel (i, j), compute a basic approximation of the gradient magnitude:

$$\text{Gradient} = \sqrt{(u(i+1, j) - u(i-1, j))^2 + (u(i, j+1) - u(i, j-1))^2}$$

- **Calculate Diffusivity:** Based on the gradient, define diffusivity $g(\text{gradient})$ as:

$$g(\text{gradient}) = \frac{1}{1 + (\frac{\text{gradient}}{k})^2}$$

This function decreases as the gradient increases, thus slowing diffusion near edges (high gradient regions) while allowing more diffusion in smooth areas.

4. **Final Update Rule with Edge Preservation:** Incorporating diffusivity, the final update rule for each pixel (i, j) is:

$$u(i, j)^{\text{new}} = u(i, j)^{\text{old}} + \alpha \cdot dt \cdot g(\text{gradient}) \cdot \nabla^2 u(i, j)$$

This rule combines the Laplacian with an edge-sensitive diffusivity factor to preserve important structures while smoothing the image.

Key Properties of Heat Diffusion:

- **Smoothing Effect:** Diffusion of Intensity: The heat equation models the diffusion of intensity values across the image, like the diffusion of heat in a physical system. This process results in a smoothing effect, reducing high-frequency noise.
- **Preservation of Edges and Noise Reduction:** By incorporating a spatially varying diffusion coefficient, the heat equation can selectively smooth regions of the image. In areas of low gradient (smooth regions), the diffusion coefficient is high, leading to significant smoothing. Conversely, in areas of high gradient (edges), the diffusion coefficient is low, preserving edge details. This adaptive smoothing behavior enables the heat equation to balance noise reduction and edge preservation effectively.
- **Flexibility:** The diffusion coefficient can be adjusted to control the degree of smoothing and edge preservation, allowing for customization to specific image characteristics and diagnostic needs.
- **Feature Enhancement:** The modified heat equation can amplify subtle features, such as small variations in the spine curvature, making them more easily detectable by automated detection algorithms.
- **Computational Efficiency:** Compared to some other advanced image processing techniques, the modified heat equation often requires less computational resources, making it suitable for real-world applications.

5.2. Anisotropic Diffusion: A Powerful Pre-processing Technique

Anisotropic diffusion is a non-linear filtering technique that selectively smooths image regions while preserving important edges. This property makes it particularly suitable for pre-processing X-ray images for scoliosis detection. By reducing noise in homogeneous regions, anisotropic diffusion enhances the visibility of subtle features like vertebral boundaries, which are crucial for accurate curvature analysis.

Mathematically, the Anisotropic Diffusion is expressed as [\[30\]](#):

$$\frac{\partial I_t(x,y)}{\partial t} = \nabla \cdot (c(|\nabla I_t|) \nabla I_t)$$

Where:

- $I_t(x, y)$ is the image intensity at position (x, y) and time t .
- ∇ is the gradient operator.
- $c(|\nabla I_t|)$ is a diffusion coefficient that depends on the gradient magnitude.

Implementation Steps:

1. **Gradient Calculation:** Gradients along both axes are computed using finite differences:

- For horizontal gradients:

$$grad_x = u(x + 1, y) - u(x-1, y)$$

- For vertical gradients:

$$grad_y = u(x, y + 1) - u(x, y-1)$$

2. **Gradient Magnitude:** The magnitude of these gradients is calculated as:

$$grad\ magnitude = \sqrt{(grad_x)^2 + (grad_y)^2}$$

3. **Diffusivity Function:** An edge-sensitive diffusivity function is computed based on the gradient magnitude:

$$c(grad\ magnitude) = e^{-\frac{(grad\ magnitude)^2}{2k^2}}$$

This function determines how much smoothing will occur based on local edge strength; areas with high gradient magnitudes (edges) receive less smoothing compared to smoother regions.

4. **Divergence Calculation:** The divergence of the diffusivity-weighted gradients is calculated as follows:

- For horizontal divergence:

$$div_x = c(grad_x) \cdot grad_x$$

- For vertical divergence:

$$div_y = c(grad_y) \cdot grad_y$$

5. **Image Update:** The original image is updated using a weighted sum of these divergences:

$$u_{new} = u + \alpha \cdot dt \cdot (\nabla \cdot (div_x + div_y))$$

This update rule continues over multiple time steps until the image achieves a desirable level of smoothness with retained structural details.

Key Properties of Anisotropic Diffusion:

- **Edge Preservation:** The diffusion coefficient's behavior near edges results in a phenomenon known as backward diffusion. This effectively enhances edges rather than blurring them.
- **Noise Reduction:** In regions with low gradients (e.g., homogeneous areas), the diffusion coefficient allows for forward diffusion, which helps to smooth out noise.
- **Convergence Speed:** The convergence speed of the diffusion coefficient plays a crucial role in determining the level of edge preservation. A higher convergence speed generally leads to better edge preservation.

Anisotropic diffusion prepares the X-ray images through this controlled diffusion process, creating an optimal balance between clarity and structural detail, essential for subsequent scoliosis detection methods.

5.3. Classification Stage using CNN [49]

Convolutional Neural Networks (CNNs) are a class of deep learning models specifically designed to process grid-like data, such as images and time series. They have been pivotal in achieving breakthroughs in tasks involving visual data, such as image recognition and computer vision, by efficiently learning spatial hierarchies of features. CNNs are composed of layers like convolutional, pooling, and fully connected layers, which allow them to automatically capture patterns such as edges, textures, and shapes in the input data. This makes them particularly effective for tasks like image classification and object detection.

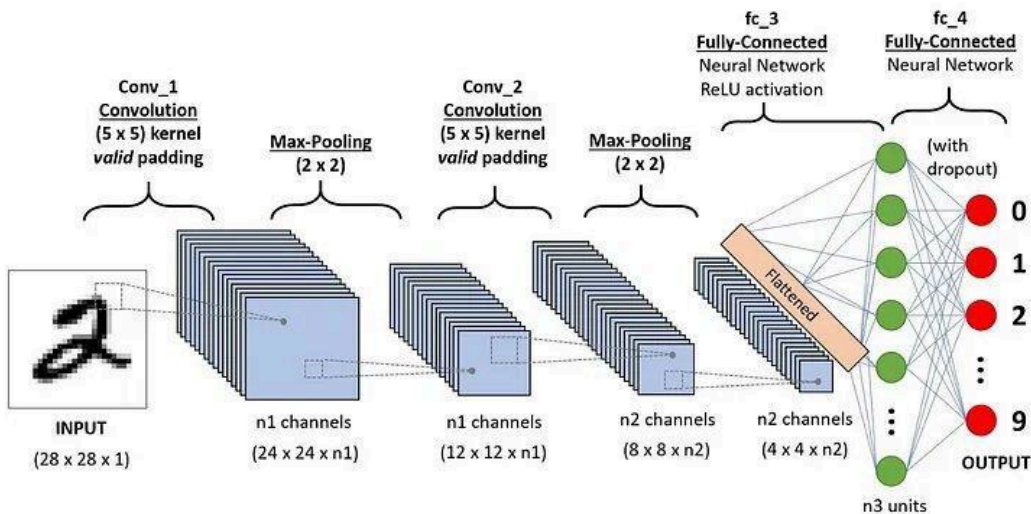


Figure 5.3: Convolutional Neural Network (CNN) architecture depicting data flow through convolution, max-pooling, and fully connected layers for image classification.

5.3.1. Model Architecture

i. Convolutional Layer

The convolutional layer is the core building block of a CNN. It performs a mathematical operation called convolution, which involves sliding a filter (or kernel) across the input image and computing dot products between the filter and local regions of the image. This process helps the network learn local patterns, such as edges, corners, and textures. Filters, which are small matrices like 3x3 or 5x5, are trained to detect specific patterns, and multiple filters are often used in each layer to capture diverse features. Parameters like stride (the step size of the filter) and padding (adding extra pixels around the image to preserve spatial dimensions) further refine the operation.

$$\text{Feature Map}(i, j) = \sum_{p=1}^k W(p) \cdot (\text{Pixel Value})_p + b$$

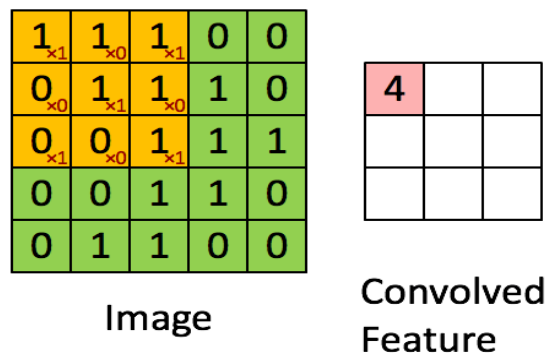


Figure 5.3.1(i): Convolution operation in a CNN showing an input image matrix and the resulting convolved feature matrix after applying a kernel.

ii. Kernel

A kernel, also known as a filter, is a small matrix that plays a crucial role in the convolution operation. This matrix slides across the input data, such as an image, and performs a dot product with the local region of the input to extract features. Kernels are typically small, such as 3x3 or 5x5, and are designed to detect specific patterns like edges, corners, or textures. Each kernel in a CNN learns to recognize a unique pattern, with multiple kernels in a layer capturing diverse features from the input data. The kernel's ability to focus on local regions and share weights across the entire input makes the network efficient and effective in extracting meaningful patterns.

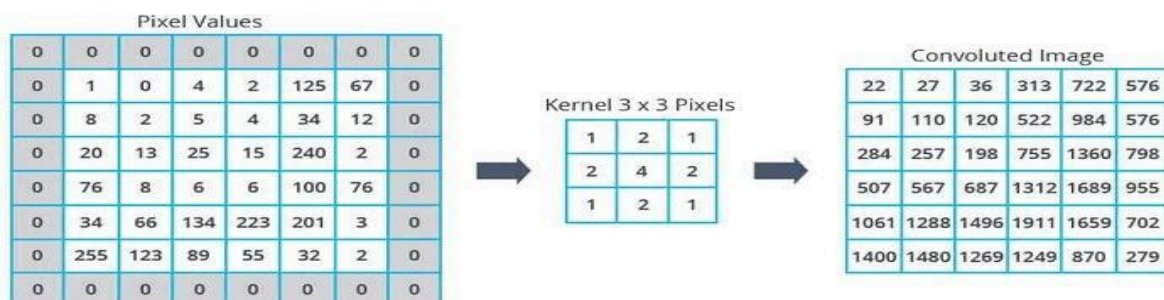


Figure 5.3.1(ii): Convolution operation using a 3x3 kernel on a 6x6 pixel image, demonstrating feature extraction in CNNs.

iii. Activation Function (ReLU)

An activation function is applied after the convolution operation to introduce non-linearity into the model. A commonly used activation function in CNNs is the Rectified Linear Unit (ReLU), which replaces all negative values in the feature map with zero. This simple operation allows the network to learn complex patterns by enabling it to model non-linear relationships within the data.

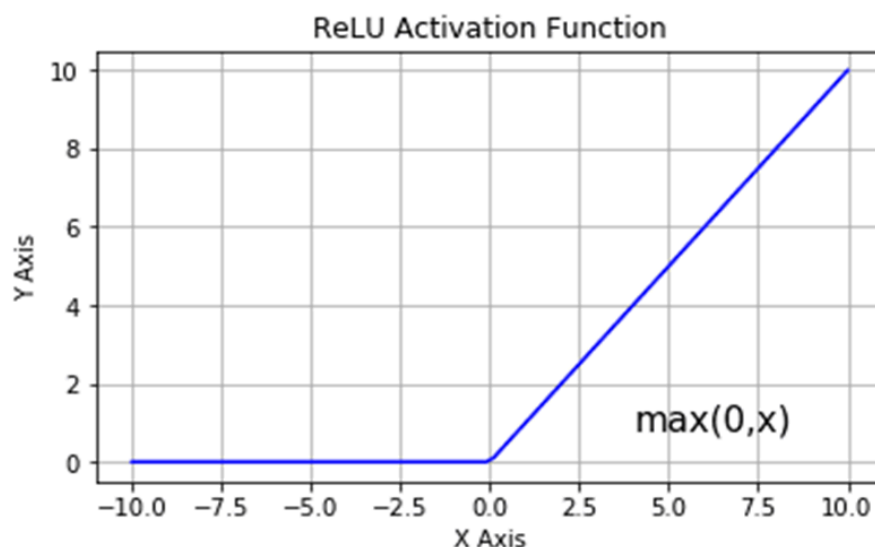


Figure 5.3.1(iii): ReLU Activation Function graph showing the function $\max(0, x)$ with an x-axis from -10 to 10 and a y-axis from 0 to 10.

iv. Pooling Layer (Max Pooling)

Pooling layers serve to downsample feature maps, reducing their spatial dimensions while retaining the most important information. Max pooling, a widely used operation, selects the maximum value from each local region of the feature map. This reduces computational complexity and makes the network more robust to variations like small translations of objects in the input image. The reduced size of the feature maps also helps mitigate overfitting.

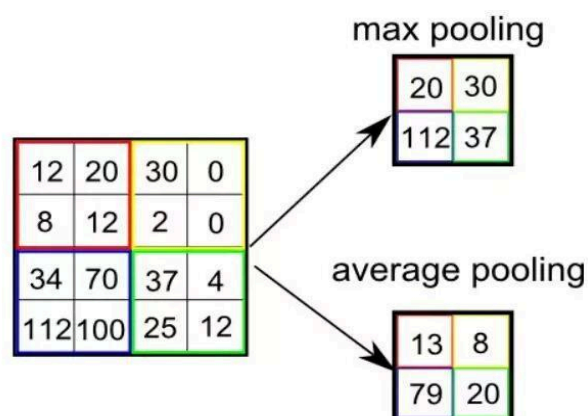


Figure 5.3.1(iv): Comparison of max pooling and average pooling operations on a sample matrix in neural networks.

v. Fully Connected (Dense) Layer

Following the convolutional and pooling layers, the feature maps are flattened into a one-dimensional vector and passed into fully connected layers. These layers, composed of neurons connected to every node in the previous layer, perform the classification task by combining the learned features. The final layer typically employs a softmax or sigmoid activation function to generate class probabilities, producing the model's output. For instance, in a medical imaging scenario, this could be classifying an image as "Normal" or "Scoliosis."

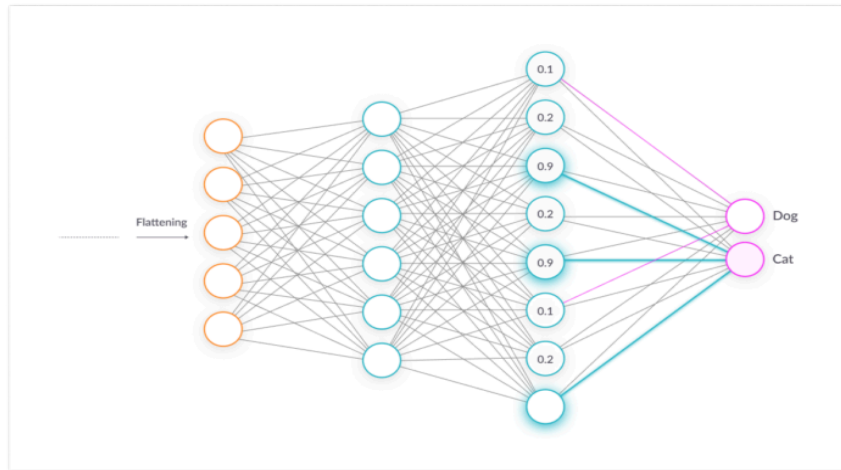


Figure 5.3.1(v): Illustration of a fully connected (dense) layer in a neural network, showing the process of flattening feature maps and classifying outputs.

vi. Forward Propagation in CNNs

The forward propagation process begins with feeding the input (e.g., a grayscale image with dimensions 25 x 50 pixels) into the CNN. Convolutional layers apply filters to detect low-level features like edges, followed by an activation function like ReLU to introduce non-linearity. Pooling layers then downsample the feature maps, retaining critical information and improving computational efficiency. The feature maps are flattened and passed through fully connected layers for classification, with the final layer providing probabilities for each class.

The computation at each layer L can be expressed as:

$$Z^{[L]} = W^{[L]} \cdot A^{[L-1]} + b^{[L]}$$

$$A^{[L]} = g(Z^{[L]})$$

Where:

- $Z^{[L]}$: Linear combination of weights and inputs
- $W^{[L]}$: Weight matrix
- $b^{[L]}$: Bias term
- $A^{[L]}$: Activation output
- $g(Z)$: Activation function (e.g., ReLU)

vii. Backward Propagation and Model Training

Backward propagation, or backpropagation, is the process used to train the CNN. It starts with calculating the loss between the predicted output and the true label using a loss function, such as categorical cross-entropy. The gradients of the loss function with respect to the model's weights are computed using the chain rule of calculus. These gradients are used to adjust the weights via optimization algorithms like Stochastic Gradient Descent (SGD) or Adam. This process is repeated over multiple iterations (epochs), gradually improving the model's accuracy by minimizing the loss.

$$\frac{\partial E}{\partial W_{ij}} = \frac{\partial E}{\partial a_i} \cdot \frac{\partial a_i}{\partial Z} \cdot \frac{\partial Z}{\partial W_{ij}}$$

Where:

- $\frac{\partial E}{\partial W_{ij}}$: Gradient for weight update
- a_i : Activation output of node i

Weights are updated using gradient descent:

$$W_{\text{new}} = W_{\text{old}} - \alpha \cdot \frac{\partial E}{\partial W}$$

Here, α is the learning rate.

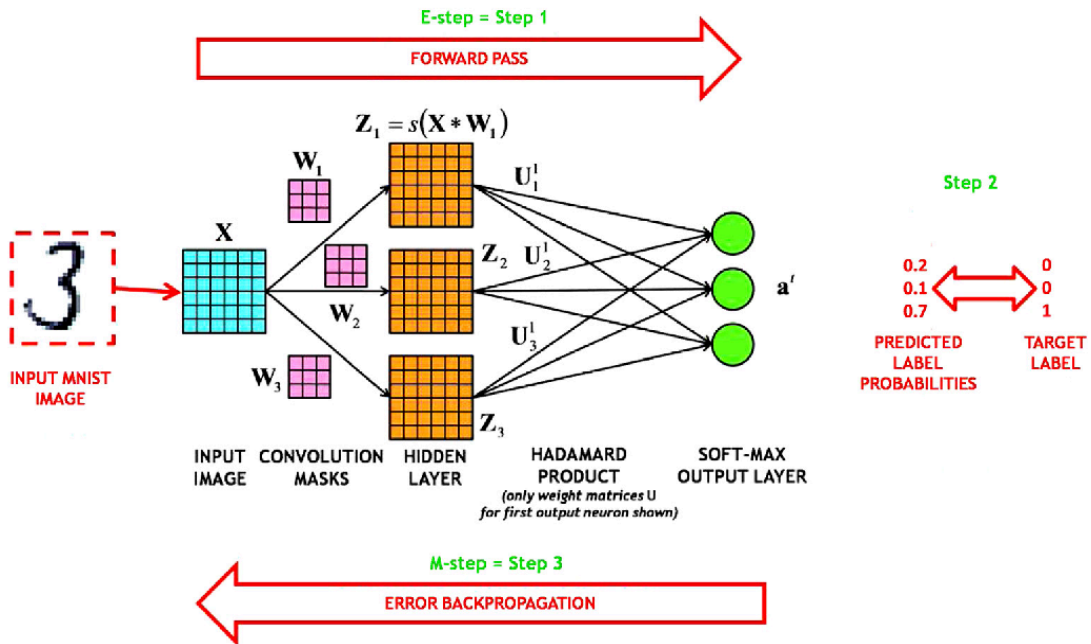


Figure 5.3.1(vii): CNN architecture and process for image classification, illustrating forward pass, convolutional filters, hidden layers, softmax output, and backward propagation.

5.4. U-Net for Binary Segmentation [53],[54]

The U-Net architecture, a fully convolutional network, was implemented for the binary segmentation of spinal cords from grayscale images. The model's symmetric architecture consists of an encoder-decoder structure, optimized for pixel-level segmentation tasks. Skip connections were employed to retain spatial details lost during downsampling, facilitating precise segmentation of anatomical features such as the spinal cord.

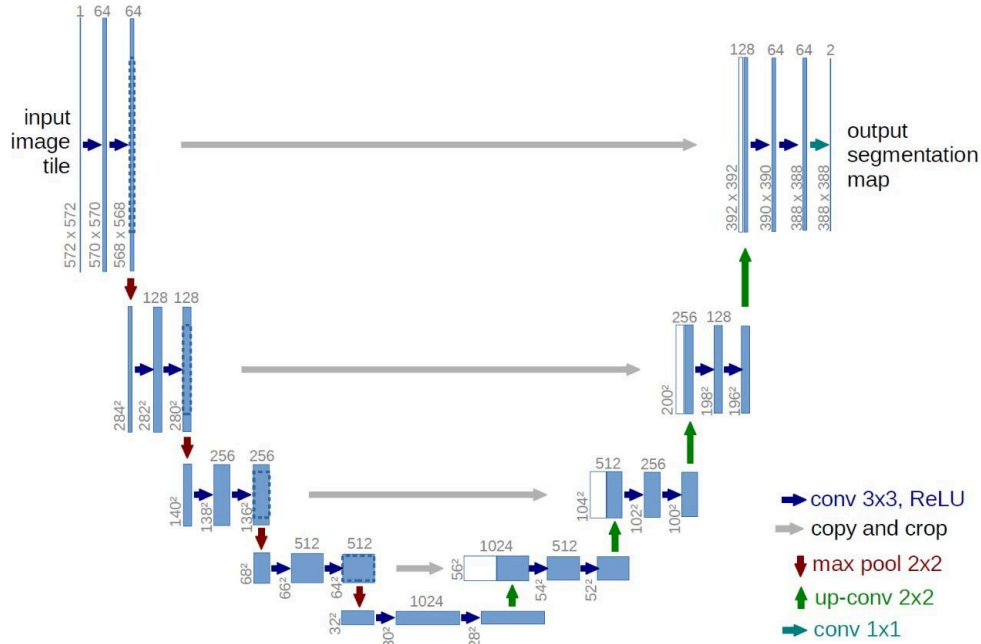


Figure 5.4: U-Net architecture for image segmentation, showing the encoder (downsampling), bottleneck, and decoder (upsampling) paths with skip connections for precise localization.

5.5. Cobb Angle Calculation Using Automated Image Analysis

The Cobb angle is a vital metric for diagnosing and monitoring scoliosis, defined as the angle between the most tilted vertebrae at the apex of the curve. Traditionally measured manually, this process is prone to variability and subjectivity. This section details an automated, computational approach for Cobb angle determination using Python-based algorithms. By combining image processing, interpolation, and vector analysis, the method ensures precise, consistent, and efficient assessment of spinal curvature, offering significant advantages for clinical and research applications.

i. Preprocessing:

A grayscale image of the spine's contour is read and inverted to highlight the spinal structure against a dark background. Contour extraction is performed using the `cv2.findContours` function from OpenCV. The largest connected contour, representing the spinal column, is selected for further processing.

ii. Smoothing the Centerline:

To mitigate noise and irregularities, the contour's centerline is smoothed using cubic spline interpolation with the `UnivariateSpline` function. The smoothing parameter s is optimized to balance the curve's fidelity to the original data and its smoothness. The criterion for selecting the optimal s includes minimizing curvature while ensuring the smoothed curve stays within a predefined drift limit from the original contour.

iii. Sampling Key Points:

Nineteen evenly spaced points are sampled along the smoothed centerline. These points provide a representation of the curve's geometry while maintaining computational efficiency.

iv. Tangent and Normal Vector Calculation:

The slope of the tangent at each sampled point is computed using the first derivative of the spline. Normal vectors, which are perpendicular to these tangents, are derived algebraically, ensuring they are unit vectors.

v. Identifying Key Points for Cobb Angle:

The points with maximum and minimum slopes are identified, corresponding to the apex and the inflection points of the spinal curve. These points define the orientation of the most tilted segments of the spinal column.

vi. Intersection of Normal Vectors:

The normal vectors at the points of maximum and minimum slopes are extended as lines. The intersection of these lines represents the focal point of angular deviation, crucial for calculating the Cobb angle.

vii. Angle Calculation:

The angle between the normal vectors is determined using the dot product formula, ensuring the calculation yields the acute angle. The Cobb angle is then computed as θ , which aligns with the clinical definition.

6. Experimental Work

6.1. Data Description

i. Dataset Overview

The dataset comprises grayscale spinal X-ray images categorized as "Normal" and "Scoliosis," resized to 25×50 pixels for standardization. It includes over 580 normal and 580 scoliosis scans, sourced from Egyptian clinics and open datasets. Clinics such as **ARC for Scoliosis Physiotherapy** (Dr. Mahmoud Ibrahim, PhD, PT), **ScolioCare** (Dr. Sarah M. Ali, MSc, PT), **Qawam for Spinal Rehabilitation** (Dr. Tayseer Saber Abdeldayem), and **4Kids Therapy Clinic** (Dr. Sarah M. Ali, MSc, PT) contributed real-world data. Open datasets like the **Mendeley Dataset** ([Link](#)) and **Roboflow Universe** ([Link](#)) provided diverse samples. This mix supports the development of a robust, generalizable scoliosis detection model. The dataset was divided into training and validation subsets, with 10% of the data reserved for validation.

ii. Data Preprocessing

- **Grayscale Conversion:** Ensures uniformity as the images originally contained varying color formats.
- **Resizing:** Standardized to 25×50 pixels.
- **Normalization:** Pixel values were scaled to the range $[0, 1]$ to stabilize training.

6.2. Classification Model Architecture [\[46\]](#)

i. Input Layer

- **Input Shape:** $25 \times 50 \times 1$ (grayscale image).
- **Purpose:** Accepts preprocessed grayscale images and feeds them into the network.

ii. First Convolutional Layer

- **Layer Type:** **Conv2D**
- **Number of Filters:** 50
- **Kernel Size:** 3×3
- **Activation Function:** ReLU (Rectified Linear Unit).
- **Features Learned:** Detects low-level features like edges and corners.
- **Input Shape:** $125 \times 50 \times 1$
- **Output Shape:** $23 \times 48 \times 50$

iii. First Pooling Layer

- **Layer Type:** **MaxPooling2D**
- **Pool Size:** 2×2
- **Purpose:** Reduces spatial dimensions by selecting the maximum value in each 2×2 region.
- **Output Shape:** $11 \times 24 \times 50$

iv. Second Convolutional Layer

- **Layer Type:** Conv2D
- **Number of Filters:** 100
- **Kernel Size:** 3×3
- **Activation Function:** ReLU.
- **Features Learned:** Captures more complex patterns and textures.
- **Output Shape:** 9×22×100

v. Second Pooling Layer

- **Layer Type:** MaxPooling2D
- **Pool Size:** 2×2
- **Purpose:** Further reduces spatial dimensions while retaining critical features.
- **Output Shape:** 4×11×100

vi. Third Convolutional Layer

- **Layer Type:** Conv2D
- **Number of Filters:** 100
- **Kernel Size:** 3×3
- **Activation Function:** ReLU.
- **Features Learned:** Extracts high-level abstract features.
- **Output Shape:** 2×9×100

vii. First Dense Layer

- **Layer Type:** Flatten
- **Purpose:** Converts the 3D tensor into a 1D vector for input into dense layers.
- **Output Shape:** 1800

viii. First Dense Layer

- **Layer Type:** Dense
- **Number of Neurons:** 100
- **Activation Function:** ReLU.

ix. Output Layer

- **Layer Type:** Dense
- **Number of Neurons:** 2
- **Activation Function:** Softmax.
- **Purpose:** Output probabilities for the two classes ("Normal" and "Scoliosis").
- **Purpose:** Learns high-level combinations of extracted features.

6.3. Training Methodology

i. Dataset Splitting

The dataset was divided into:

- **Training Set:** 80% of the data.
- **Validation Set:** 10% of the data.
- **Testing Set:** 10% of data.

ii. Hyperparameters

- **Batch Size:** 50.
- **Number of Epochs:** 20.
- **Optimizer:** Adam.
- **Loss Function:** Sparse Categorical Cross Entropy
- **Evaluation Metric:** Accuracy

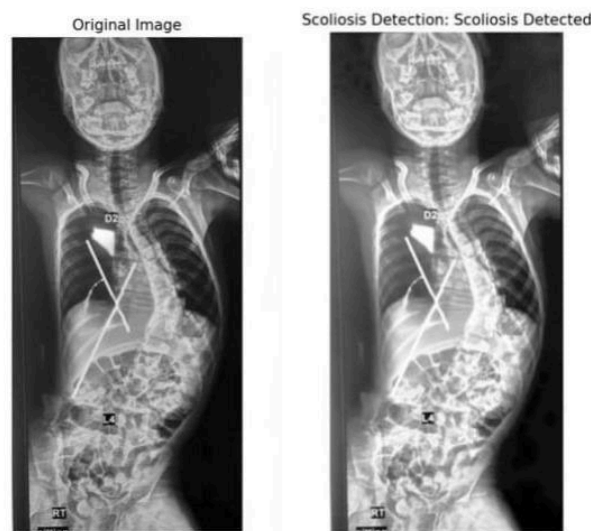


Figure 6.3: Performance of CNN Classification Model

The model was trained using TensorFlow's `model.fit()` method, which tracked both training and validation performance.

6.4. Line Segmentation Using U-net [\[53\],\[54\]](#)

The U-Net model segments the spinal X-ray images into a binary mask, where the foreground (spinal cord) is separated from the background. The segmentation process is mathematically expressed as follows:

i. Input:

The input layer takes grayscale spinal images with dimensions $256 \times 256 \times 1$. These images are normalized to have pixel values between 0 and 1 to ensure uniformity and stable training.

ii. Encoding Path:

The encoder consists of a series of convolutional and max-pooling layers. Each block begins with two 3×3 convolutional layers with ReLU activation, extracting low- to high-level features. Max-pooling layers (2×2) are applied to reduce the spatial dimensions by half while increasing the depth of the feature maps (e.g., $64 \rightarrow 128 \rightarrow 256 \rightarrow 512$). Dropout layers are incorporated to prevent overfitting by randomly deactivating a portion of neurons.

iii. Bottleneck:

At the center of the network, the bottleneck captures the most abstract and high-level features. It consists of two 3×3 convolutional layers with 1024 filters and ReLU activation, followed by a dropout layer to ensure generalization.

iv. Decoding Path:

The decoder reconstructs the segmentation map's spatial resolution using upsampling operations. Transposed convolutional layers (2×2) double the dimensions of feature maps, while skip connections concatenate feature maps from corresponding encoder layers. This fusion of high-level and low-level features preserves fine-grained spatial details. Each block also includes two 3×3 convolutional layers with ReLU activation to refine features.

v. Output Layer:

The final layer uses a 1×1 convolution to map the feature maps into a single-channel segmentation mask. A sigmoid activation function is applied to produce pixel-wise probabilities, where each pixel is classified as either spinal cord (foreground) or background.

vi. Training Process:

During training, the model optimizes a binary cross-entropy loss function, which measures the difference between predicted and true segmentation masks. The loss function's gradients are computed through backpropagation and updated using an optimization algorithm like **Adam**. The training process iteratively minimizes the loss to achieve accurate segmentation results.

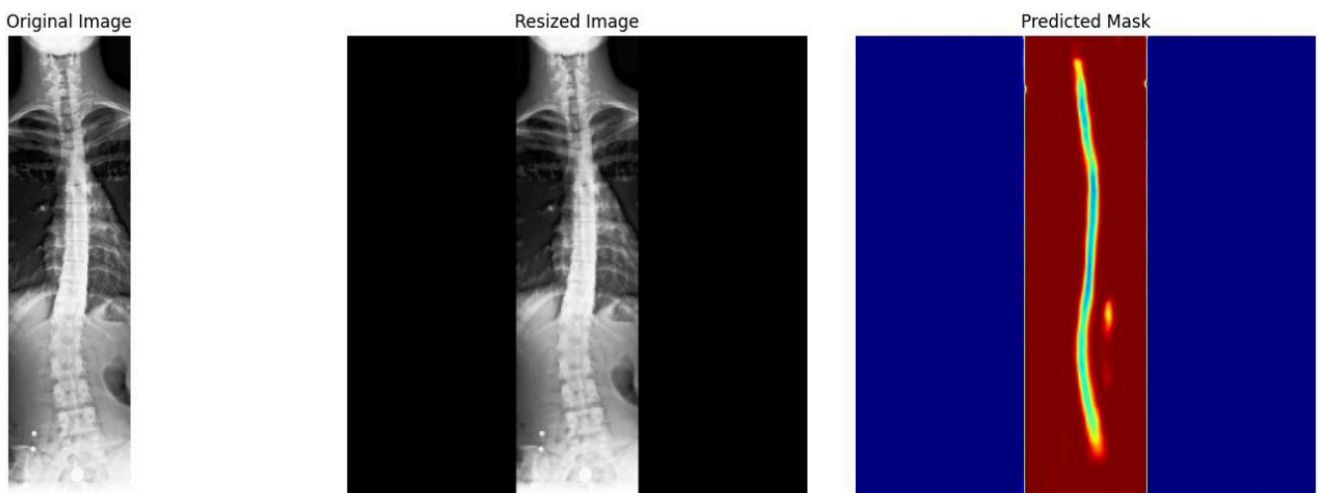


Figure 6.4: Performance of Line Segmentation using U-Net

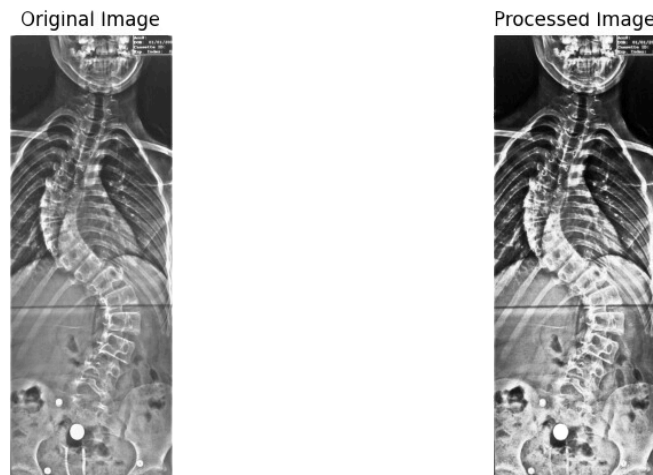
7. Results

7.1. Comparison of Heat Equation and Anisotropic Diffusion for Pre-processing Spinal X-ray Images

In this study, we evaluated the effectiveness of the heat equation and anisotropic diffusion for pre-processing spinal X-ray images. Both methods were applied to a dataset of noisy X-ray images, and their performance was assessed in terms of noise reduction, edge preservation, and overall image quality.

Heat Equation

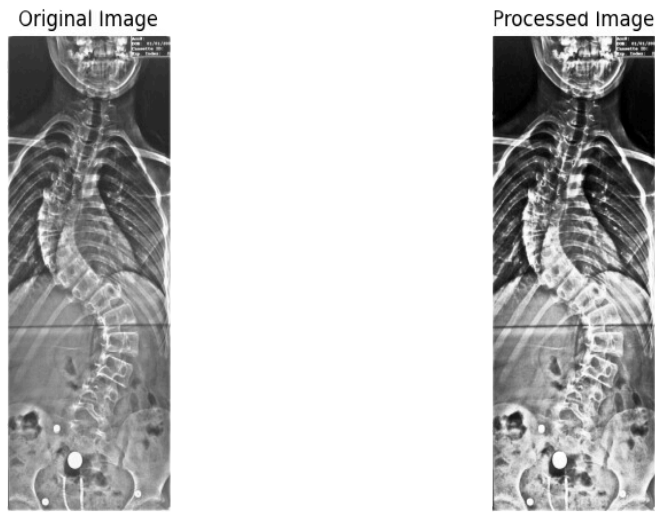
- **Noise Reduction:** The heat equation effectively reduced noise in the images, particularly in regions with low-frequency noise. However, it also tended to blur edges, especially in regions with high-frequency noise.
- **Edge Preservation:** The heat equation's isotropic diffusion nature limited its ability to preserve sharp edges. While it could reduce noise, it often compromised the fine details of the spinal structures.



7.1(1)Heat Equation Pre-processing Result

Anisotropic Diffusion

- **Edge Preservation:** Anisotropic diffusion demonstrated superior edge preservation capabilities. By adaptively adjusting the diffusion coefficient based on the image gradient, it effectively prevented the blurring of edges, especially in regions with high-frequency noise.
- **Noise Reduction:** While it was effective in reducing noise, particularly in homogeneous regions, anisotropic diffusion may not be as effective as the heat equation in reducing low-frequency noise.












7.1(2)Anisotropic Diffusion Pre-processing Result

Experimental results and discussion

In this study, we evaluated the effectiveness of the heat equation and anisotropic diffusion for pre-processing spinal X-ray images. Both methods were applied to a dataset of noisy X-ray images, and their performance was assessed in terms of noise reduction, edge preservation, and overall image quality.

Overall Comparison

<i>Noisy Image</i>	<i>Heat Equation</i>	<i>Anisotropic Diffusion</i>
		
		
		

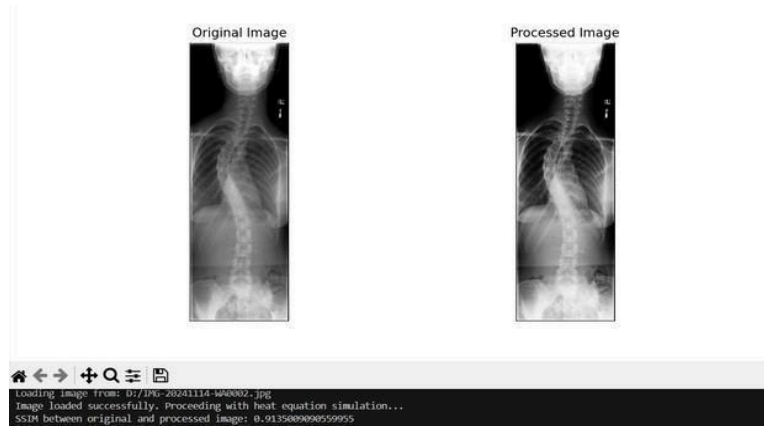
7.2. Structural Similarity Index Measurements (SSIM) [47]:

$$SSIM(I_o, I_r) = \frac{(2\mu_x\mu_y + \alpha)(2cov(I_o, I_r) + \beta)}{(\mu_x^2 + \mu_y^2 + \alpha)(\sigma_x^2 + \sigma_y^2 + \beta)}$$

where μ_x and μ_y are the means and σ_x and σ_y are the variances of the images I_o and I_r respectively. $cov(I_o, I_r)$ is the covariance of the two images, and α, β are small values, used to stabilize the division with small denominators. SSIM is used to show the similarity of the structure of the input image to the structure of the restored image. The SSIM lies between 0 and 1. The higher value of SSIM shows a great structural similarity between the input and restored image.

SSIM Result for Heat Equation

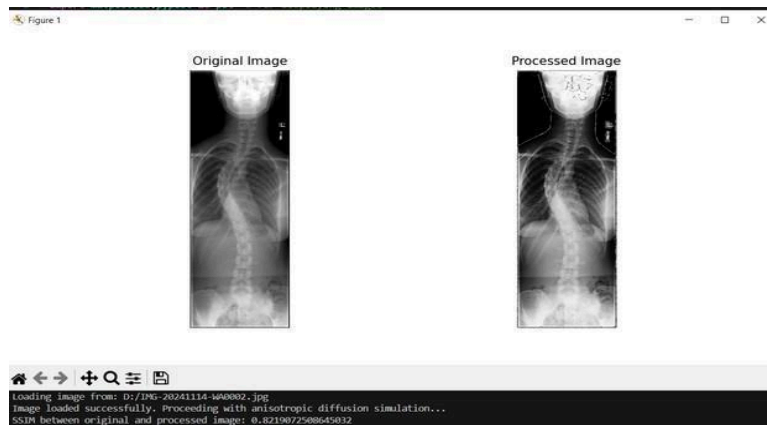
The heat equation resulted in a Structural Similarity Index Measurements (SSIM) score of 0.9135, indicating good perceptual quality.



7.2(1): SSIM Result for Heat Equation

SSIM Result for Anisotropic Diffusion:

The anisotropic diffusion resulted in a Structural Similarity Index (SSIM) score of 0.8219, indicating a lower level of perceptual similarity.



7.2(2): SSIM Result for Anisotropic Diffusion

7.3. Results of the CNN model's performance

Our model was trained for 20 epochs, achieving a training accuracy of **96.64%**, a validation accuracy of **98.91%**, and a test accuracy of **84.58%**.

```
17/17 - 1s - 42ms/step - accuracy: 0.8657 - loss: 0.3207 - val_accuracy: 0.9239 - val_loss: 0.2780
Epoch 7/20
17/17 - 1s - 42ms/step - accuracy: 0.9029 - loss: 0.2552 - val_accuracy: 0.9565 - val_loss: 0.1708
Epoch 8/20
17/17 - 1s - 42ms/step - accuracy: 0.9065 - loss: 0.2280 - val_accuracy: 0.8913 - val_loss: 0.1847
Epoch 9/20
17/17 - 1s - 44ms/step - accuracy: 0.9077 - loss: 0.2374 - val_accuracy: 0.9239 - val_loss: 0.1731
Epoch 10/20
17/17 - 1s - 44ms/step - accuracy: 0.9113 - loss: 0.2059 - val_accuracy: 0.9783 - val_loss: 0.0976
Epoch 11/20
17/17 - 1s - 45ms/step - accuracy: 0.9245 - loss: 0.1870 - val_accuracy: 0.9239 - val_loss: 0.2094
Epoch 12/20
17/17 - 1s - 43ms/step - accuracy: 0.9257 - loss: 0.1838 - val_accuracy: 0.9783 - val_loss: 0.0666
Epoch 13/20
17/17 - 1s - 48ms/step - accuracy: 0.9664 - loss: 0.1168 - val_accuracy: 0.9891 - val_loss: 0.0371
Epoch 14/20
17/17 - 1s - 48ms/step - accuracy: 0.9424 - loss: 0.1417 - val_accuracy: 0.9783 - val_loss: 0.0831
Epoch 15/20
17/17 - 1s - 46ms/step - accuracy: 0.9245 - loss: 0.1999 - val_accuracy: 0.9783 - val_loss: 0.1186
Epoch 16/20
17/17 - 1s - 42ms/step - accuracy: 0.9424 - loss: 0.1682 - val_accuracy: 0.9457 - val_loss: 0.0932
```

Figure 7.3(1): CNN Classification Model Accuracy

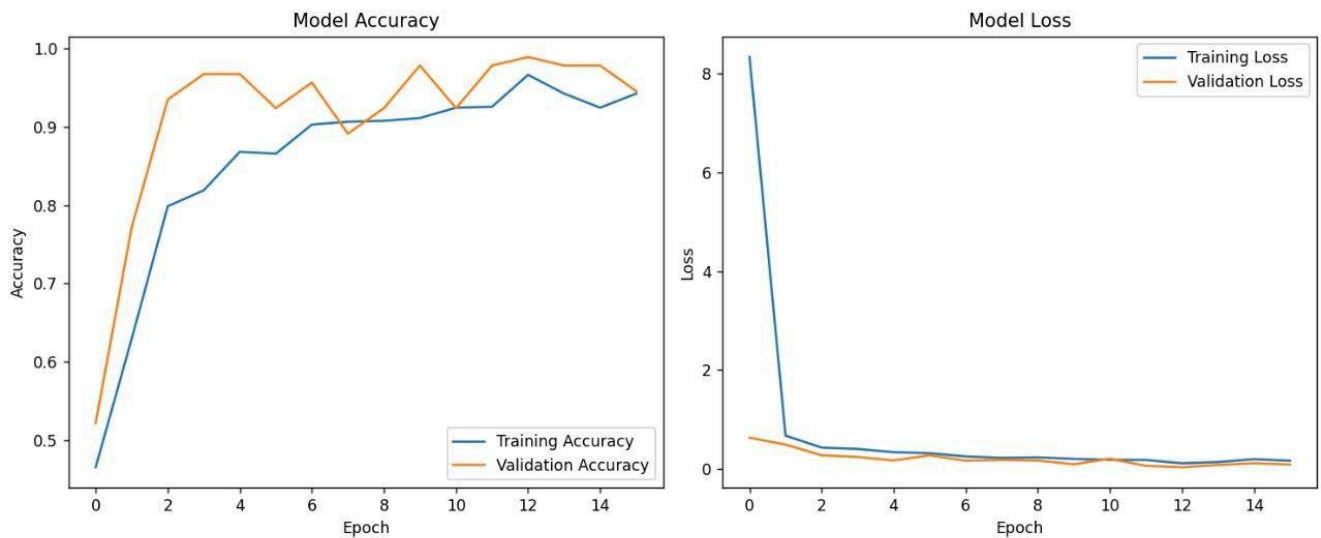


Figure 7.3(2): Performance of CNN Classification model on Training and Validation.

Testing Dataset for Scoliosis Classification

Summary statistics of the CNN model's performance on the test dataset. Out of 240 images, the model correctly predicted 203 cases, resulting in a test accuracy of **84.58%**. The model incorrectly classified 37 images, with a test loss of 0.7369, indicating a moderate level of error. These results highlight the model's effectiveness in classifying normal and scoliosis images, though there is still room for improvement.

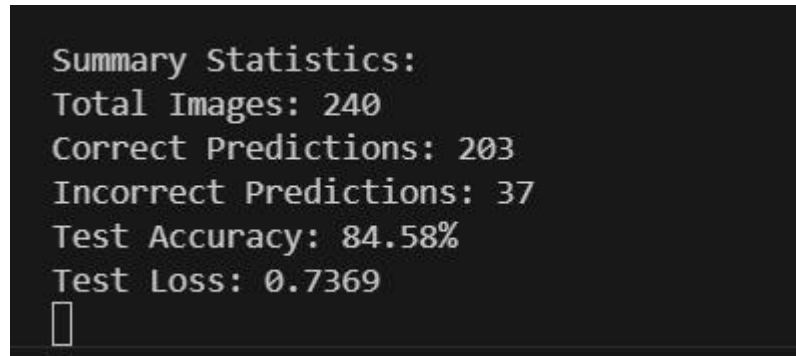


Figure 7.3(3): Testing performance of the CNN model for normal and scoliosis classification.

The bar chart illustrates the performance of the trained model on the testing dataset. The green bar represents the number of correct predictions made by the model, while the red bar represents incorrect predictions. Out of a total of 240 images in the test dataset, the model accurately classified 203 images, corresponding to a correct prediction rate of approximately **84.58%**. The remaining 37 images were misclassified, accounting for about 15.42% of the total. This visualization highlights the model's effectiveness in differentiating between the "Normal" and "Scoliosis (Scol)" classes in the test set.

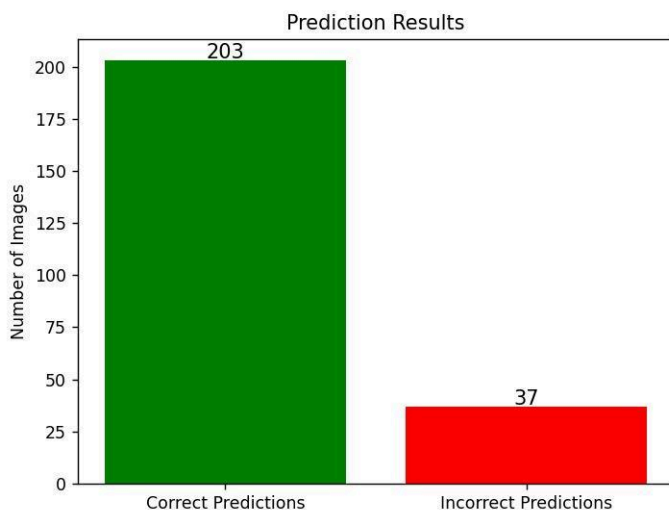


Figure 7.3(4): Bar chart showing the testing dataset

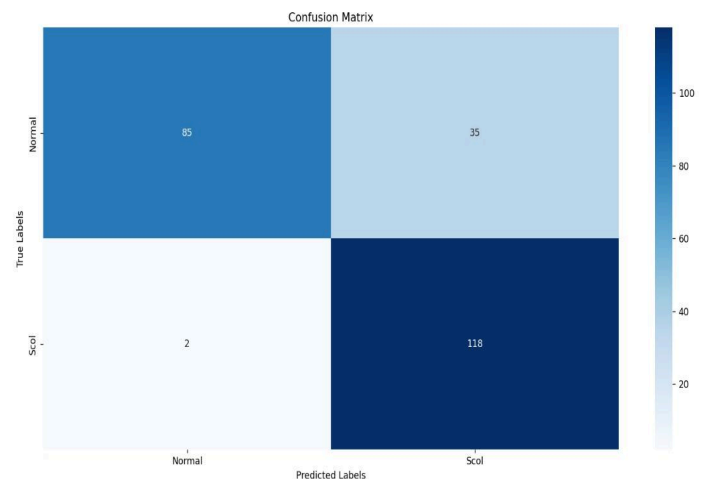


Figure 7.3(5): Confusion matrix illustrating a classification model's performance with 85 true positives, 35 false positives, 2 false negatives, and 118 true negatives.

7.4. Results of Line Segmentation Using U-Net

Our model was trained for 45 epochs, achieving a training accuracy of **99.3%**, a validation accuracy of **99.16%**, and a test accuracy of **98.74%**.

```
231
Epoch 43/50
23/23 ————— 8s 361ms/step - accuracy: 0.9928 - loss: 0.0167 - val_accuracy: 0.9916 - val_loss: 0.0
226
Epoch 44/50
23/23 ————— 8s 361ms/step - accuracy: 0.9928 - loss: 0.0166 - val_accuracy: 0.9916 - val_loss: 0.0
224
Epoch 45/50
23/23 ————— 8s 361ms/step - accuracy: 0.9930 - loss: 0.0164 - val_accuracy: 0.9916 - val_loss: 0.0
220
```

Figure 7.4(1): Line Segmentation Using U-Net Model Accuracy

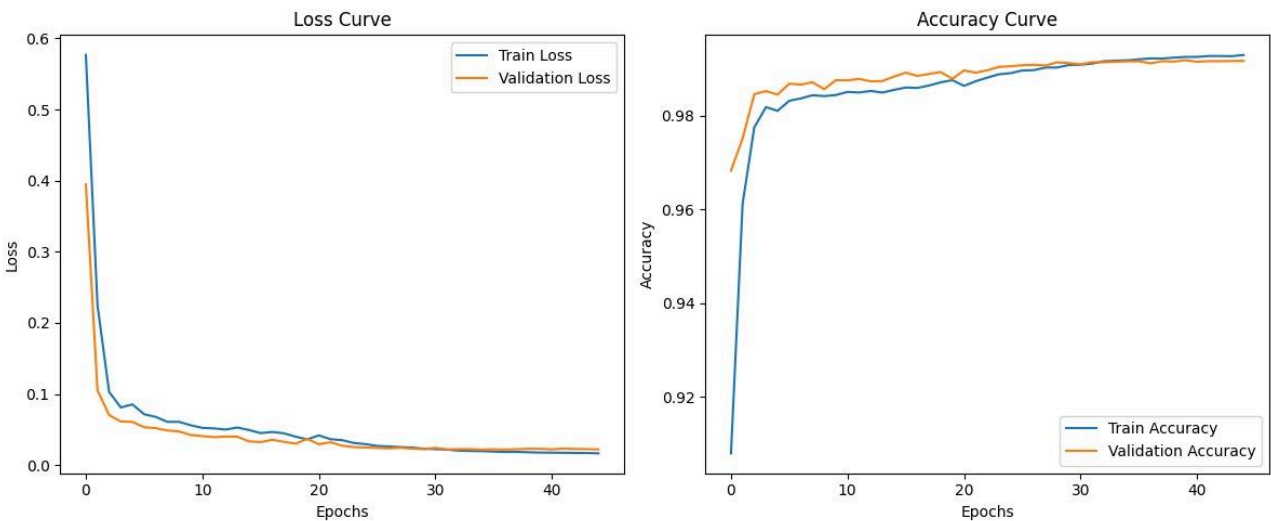


Figure 7.4(2): Performance of Line Segmentation Model on Training and Validation.

Test Cases

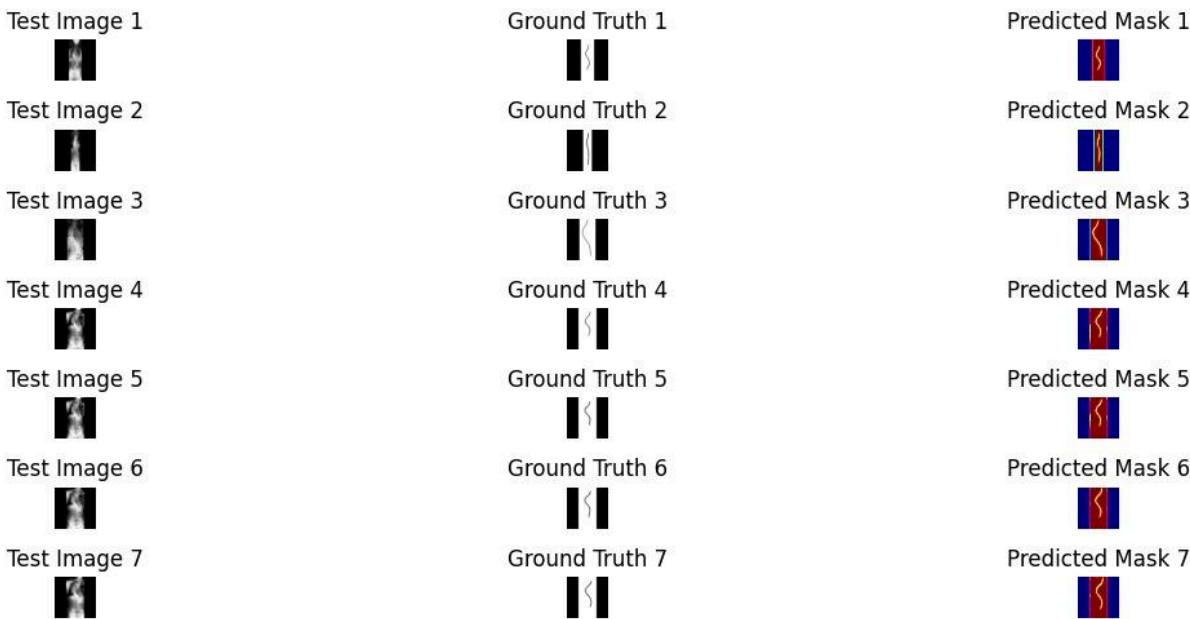
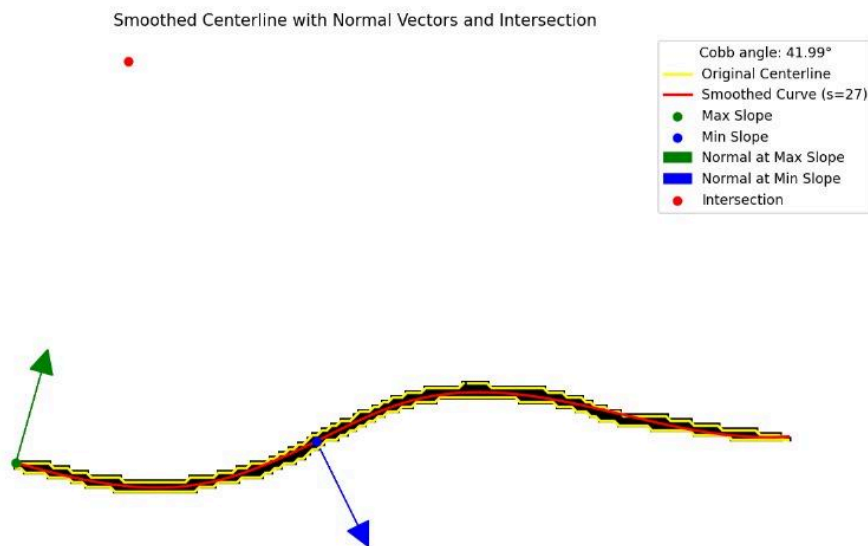


Figure 7.4(3): Test Cases of Line Segmentation

7.5. Cobb Angle Analysis and Results:

The automated algorithm was tested on X-ray images of scoliosis cases. The methodology yielded a Cobb angle of (replace with calculated value) for the provided image. The visualization includes:

- The original and smoothed spinal centerline.
- Points of maximum and minimum slopes.
- Tangents and normal vectors at these key points.
- Intersection point and corresponding Cobb angle annotation.



7.5. Smoothed spinal centerline with calculated Cobb angle (41.99°), highlighting max/min slopes, normals, and intersection.

8. Conclusion

In this study, we compared the performance of the heat equation and anisotropic diffusion for pre-processing spinal X-ray images. While anisotropic diffusion is often considered a superior technique for preserving edges and reducing noise, our results suggest that the heat equation may be more suitable for certain types of images, particularly those with lower levels of noise and less pronounced edge features. The heat equation achieved a higher SSIM score, indicating better perceptual quality. However, anisotropic diffusion may still be the preferred choice for images with significant noise and complex structures.

The results demonstrated the effectiveness of combining preprocessing techniques with a CNN for scoliosis detection, achieving high training and validation accuracies. Additionally, the integration of U-Net for spinal cord segmentation holds great promise for accurately measuring the Cobb angle, a critical metric in scoliosis diagnosis. By precisely segmenting the spinal cord, U-Net can provide more reliable landmarks for Cobb angle measurement, facilitating improved monitoring of scoliosis progression.

9. References

1. Tajdari, M., Pawar, A., Li, H., Tajdari, F., Maqsood, A., Cleary, E., Saha, S., Zhang, Y. J., Sarwark, J. F., and Liu, W. K. (2021). Image-based modeling for Adolescent Idiopathic Scoliosis: Mechanistic machine learning analysis and prediction. *Computer Methods in Applied Mechanics and Engineering*, 374, 113590. <https://doi.org/10.1016/j.cma.2020.113590>
2. Zalevsky, Z., Livshits, P., & Gur, E. (2014). New approaches to image processing based failure analysis of nano-scale ULSI devices. Elsevier. <https://doi.org/10.1016/C2013-0-06953-8>
3. Göreke, V. (2022). A novel method based on Wiener filter for denoising Poisson noise from medical X-ray images. *Biomedical Signal Processing and Control*, 79, 104031. <https://doi.org/10.1016/j.bspc.2022.104031>
4. Mohd Sagheer, S. V., and George, S. N. (2020). A review on medical image denoising algorithms. *Biomedical Signal Processing and Control*, 61, 102036. <https://doi.org/10.1016/j.bspc.2020.102036>
5. Rundo, L., Tangherloni, A., Nobile, M. S., Militello, C., Besozzi, D., Mauri, G., and Cazzaniga, P. (2019). MedGA: A novel evolutionary method for image enhancement in medical imaging systems. *Expert Systems With Applications*, 119, 387-399. <https://doi.org/10.1016/j.eswa.2018.11.013>
6. Kleinstreuer, C., Childress, E., and Kennedy, A. (2013). Targeted drug delivery: Multifunctional nanoparticles and direct micro-drug delivery to tumors. In S. M. Becker & A. V. Kuznetsov (Eds.), *Transport in biological media* (pp. 391–416). Elsevier. <https://doi.org/10.1016/B978-0-12-415824-5.00010-2>
7. Kong, W., Miao, Q., Liu, R., Lei, Y., Cui, J., and Xie, Q. (2022). Multimodal medical image fusion using gradient domain guided filter random walk and side window filtering in the framelet domain. *Information Sciences*, 585, 418-440. <https://doi.org/10.1016/j.ins.2021.11.033>
8. Ou, X., Chen, X. Xu, X., Xie, L. Chen, X., Hong, Z., Bai, H., Liu, X., Chen, Q., Li, L. (2021). Recent Development in X-Ray Imaging Technology: Future and Challenges. *Research*. DOI: [10.34133/2021/9892152](https://doi.org/10.34133/2021/9892152)
9. Takahide, O., Shigeru, F., Hiroshi, I., Shin'ya, Y., Jianming, L., Yahagi, T. (2004). Noise Reduction in Digital Radiography Using Wavelet Packet Based on Noise Characteristics. *Journal of Signal Processing*, 8. 485-494. 10.2299/jsp.8.485.
10. Nowak, R. D. and Richard B. (1999). "Wavelet-domain filtering for photon imaging systems." *IEEE transactions on image processing: a publication of the IEEE Signal Processing Society* 8 5: 666-78.
11. Wang, L., Lu, J., Li, Y., Yahagi, T., and Okamoto, T. (2005). Noise Reduction Using Wavelet with Application to Medical X-ray Image. *IEEE International Conference on Industrial Technology*. doi: [10.1109/icit.2005.1600606](https://doi.org/10.1109/icit.2005.1600606)
12. Yang, X.S., and Papa, J. P. (2016). Bio-inspired computation and applications in image processing. Elsevier Ltd. <https://doi.org/10.1016/C2015-0-00856-5>
13. Claude, K., and Zwicker, M., (2014). "Progressive image denoising." *IEEE transactions on image processing* 23, no. 7: 3114-3125
14. 1-Priyam, Ch., and Milanfar, P., (2009). "Is denoising dead?" *IEEE Transactions on Image Processing* 19(4): 895-911.
15. Kaur, H., Virmani, J., & Thakur, S. (2018). A genetic algorithm-based metaheuristic approach to customize a computer-aided classification system for enhanced screen film mammograms. *U- Healthcare Monitoring Systems*, 217-259. <https://doi.org/10.1016/B978-0-12-815370-3.00010-4>
16. Mafi, M., Martin, H., Cabrerizo, M., Andrian, J., Barreto, A., & Adjouadi, M. (2019). A comprehensive survey on impulse and Gaussian denoising filters for digital images. *Signal Processing*, 157, 236-260. <https://doi.org/10.1016/j.sigpro.2018.12.00>
17. Gupta, B., and Lamba, S. S. (2021). An efficient anisotropic diffusion model for image denoising with edge preservation. *Computers & Mathematics With Applications*, 93, 106-119. <https://doi.org/10.1016/j.camwa.2021.03.029>
18. Nakamura, S., Kobayashi, T., Funatsu, A. (2016). Patient radiation dose reduction using an X-ray imaging noise reduction technology for cardiac angiography and intervention. *Heart Vessels* 31, 655– 663 <https://doi.org/10.1007/s00380-015-0667-z>

19. ten Cate, T., van Wely, M., Gehlmann, H. (2015). Novel X-ray image noise reduction technology reduces patient radiation dose while maintaining image quality in coronary angiography. *Neth Heart J* 23, 525–530. <https://doi.org/10.1007/s12471-015-0742-1>
20. Rapp, B. E. (2017). *Microfluidics: Modeling, mechanics and mathematics*. Elsevier Inc. <https://doi.org/10.1016/C2012-0-02230-2>
21. Bovik, A. L. (2005). *Handbook of image and video processing*. Elsevier Inc. <https://doi.org/10.1016/B978-0-12-119792-6.X5062-1>
22. Frangi, A. F., Prince, J. L., and Sonka, M. (2024). *Medical image analysis*. Elsevier Ltd. <https://doi.org/10.1016/C2015-0-06316-X>
23. Wang, S., Celebi, M. E., Zhang, Y., Yu, X., Lu, S., Yao, X., Zhou, Q., Miguel, M., Tian, Y., Gorriz, J. M., & Tyukin, I. (2021). Advances in Data Preprocessing for Biomedical Data Fusion: An Overview of the Methods, Challenges, and Prospects. *Information Fusion*, 76, 376-421. <https://doi.org/10.1016/j.inffus.2021.07.001>
24. Guy, G., Sochen, N., and Zeevi, Y. Y. (2002) Forward-and-backward diffusion processes for adaptive image enhancement and denoising. *IEEE transactions on image processing* 11, (7): 689-703.
25. Pietro, P., and Malik, J. (1990) Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on pattern analysis and machine intelligence* 12, (7): 629-639.
26. Dost, R., & Kasiviswanathan, K. (2022). Analyses of drought severity and frequency in Afghanistan. *Developments in Environmental Science*, 14, 259-275. <https://doi.org/10.1016/B978-0-443-18640-0.00014-6>
27. Yuan, X., Nayoze-Coynel, C., Shaigan, N., Fisher, D., Zhao, N., Zamel, N., Gazdzicki, P., Ulsh, M., Friedrich, K. A., Girard, F., & Groos, U. (2021). A review of functions, attributes, properties and measurements for the quality control of proton exchange membrane fuel cell components. *Journal of Power Sources*, 491, 229540. <https://doi.org/10.1016/j.jpowsour.2021.229540>
28. Dimian, A. C., Bildea, C. S., & Kiss, A. A. (2013). Chemical Product Design. *Computer Aided Chemical Engineering*, 35, 489-523. <https://doi.org/10.1016/B978-0-444-62700-1.00012-7>
29. Adamo, F., Attivissimo, F., Di Nisio, A., & Savino, M. (2009). A low-cost inspection system for online defects assessment in satin glass. *Measurement*, 42(9), 1304-1311. <https://doi.org/10.1016/j.measurement.2009.05.006>
30. Gupta, B., and Lamba, S. S. (2021). An efficient anisotropic diffusion model for image denoising with edge preservation. *Computers & Mathematics With Applications*, 93, 106-119. <https://doi.org/10.1016/j.camwa.2021.03.029>
31. Bouget, D., Allan, M., Stoyanov, D., & Jannin, P. (2016). Vision-based and marker-less surgical tool detection and tracking: A review of the literature. *Medical Image Analysis*, 35, 633-654. <https://doi.org/10.1016/j.media.2016.09.003>
32. Arora, J. S. (2003). More on Numerical Methods for Unconstrained Optimum Design. *Introduction to Optimum Design (Second Edition)*, 305-337. <https://doi.org/10.1016/B978-012064155-0/50009-4>
33. Ziyang, Z., Duan, Ch., Lin, T., Zhou, Sh., Wang, Y. and Gao, X. (2020). GVFOM: a novel external force for active contour-based image segmentation. *Information Sciences* 506: 1-18
34. ,W., Ren, W., and Wang, H., (2013). Anisotropic second and fourth order diffusion models based on convolutional virtual electric field for image denoising." *Computers & Mathematics with Applications* 66(10):1729-1742.
35. Ovidiu, G., and Whelan, P. F. (2010). A new GVF-based image enhancement formulation for use in the presence of mixed noise." *Pattern Recognition* 43(8): 2646-2658.
36. Zhang, Q., Yang, L. T., Chen, Z., & Li, P. (2018). A survey on deep learning for big data. *Information Fusion*, 42, 146-157. <https://doi.org/10.1016/j.inffus.2017.10.006>
37. Yuan-Quan, W., Guo, J., Chen, W. and Zhang, W. (2013). Image denoising using modified Perona– Malik model based on directional Laplacian. *Signal Processing* 93(9): 2548-2558.
38. Mannan, S. (2013). *Computer Aids and Expert Systems*. *Lees' Process Safety Essentials*, 383-401. <https://doi.org/10.1016/B978-1-85617-776-4.00020-8>

39. Baraha, S., Sahoo, A. K., & Modalavalasa, S. (2022). A systematic review on recent developments in nonlocal and variational methods for SAR image despeckling. *Signal Processing*, 196, 108521. <https://doi.org/10.1016/j.sigpro.2022.108521>
40. Cristovao, C., Foi, A., Katkovnik, V. and Egiazarian, K. (2018). Nonlocality-reinforced convolutional neural networks for image denoising." *IEEE Signal Processing Letters* 25(8) :1216-1220.
41. Saeed, A. and Barnes, N. (2019). Real image denoising with feature attention. In *Proceedings of the IEEE/CVF international conference on computer vision*, 3155-3164.
42. Chunwei, T., Xu, Y., and Zuo, W. (2020). Image denoising using deep CNN with batch renormalization. *Neural Networks* 121: 461-473.
43. Prabhu, G.K. (2012). Automatic Quantification of Spinal Curvature in Scoliotic Radiograph using Image Processing. *J Med Syst* 36, 1943–1951. <https://doi.org/10.1007/s10916-011-9654-9>
44. Dubost, F. (2020). Automated estimation of the spinal curvature via spine centerline extraction with ensembles of cascaded neural networks. In Y. Cai, L. Wang, M. Audette, G. Zheng, and S. Li (Eds.), *Computational methods and clinical applications for spine imaging*. CSI 2019. *Lecture notes in computer science* (Vol. 11963). Springer, Cham. https://doi.org/10.1007/978-3-030-39752-4_10
45. Tajdari, M., Pawar, A., Li, H., Tajdari, F., Maqsood, A., Cleary, E., Saha, S., Zhang, Y. J., Sarwark, J. F., & Liu, W. K. (2021). Image-based modelling for Adolescent Idiopathic Scoliosis: Mechanistic machine learning analysis and prediction. *Computer Methods in Applied Mechanics and Engineering*, 374, 113590. <https://doi.org/10.1016/j.cma.2020.113590>
46. Claudio, V., Wafa, S., and Laurent, G. (2020). A convolutional neural network to detect scoliosis treatment in radiographs. *International Journal of Computer Assisted Radiology and Surgery*. 15. 10.1007/s11548-020-02173-4.
47. Bakurov, I., Buzzelli, M., Schettini, R., Castelli, M., & Vanneschi, L. (2022). Structural similarity index (SSIM) revisited: A data-driven approach. *Expert Systems With Applications*, 189, 116087. <https://doi.org/10.1016/j.eswa.2021.116087>
48. Frangi, A. F., Prince, J. L., & Sonka, M. (2024). *Medical Image Analysis*. Elsevier Ltd. <https://doi.org/10.1016/C2015-0-06316-X>
49. U. Kose, D. Gupta, D. A. V. H. C., and A. Khanna, "Chapter 7 - Deep convolutional neural network-based image classification for COVID-19 diagnosis," in *Data Science for covid-19*, Amsterdam: Academic Press, 2021.
50. Raju, I., Banu, M. S., Mim, S. A., Hossain, S., & Saha, H. K. (2022). A case study on simulation of heat equation by Crank-Nicolson method in accordance with digital image processing. *International Journal of Scientific & Engineering Research*, 13(1). <https://doi.org/ISSN2229-5518>
51. Effectiveness of the Boston Brace in the Treatment of Paediatric Scoliosis: A Longitudinal Study from 2010–2020 in a National Spinal Centre. <https://www.mdpi.com/2227-9032/11/10/1491>
52. MILWAUKEE BRACE TREATMENT OF SCOLIOSIS ,JOHN E. LONSTEIN, Clinical Professor, Department of Orthopedic Surgery, University of Minnesota, Minneapolis, Minnesota. Twin Cities Spine Center, Minneapolis, Minnesota. https://www.srs.org/Files/Research/Manuals-and-Publications/SRS_Milwaukee_Brace.2003.pdf.
53. Liu, J., Yuan, C., Sun, X. *et al.* The measurement of Cobb angle based on spine X-ray images using multi-scale convolutional neural network. *Phys Eng Sci Med* 44, 809–821 (2021). <https://doi.org/10.1007/s13246-021-01032-z>
54. Horng, H., Kuok, P., Fu, J., Lin, J., & Sun, N. (2018). Cobb Angle Measurement of Spine from X-Ray Images Using Convolutional Neural Network. *Computational and Mathematical Methods in Medicine*, 2019(1), 6357171. <https://doi.org/10.1155/2019/6357171>

Link to our Repository on Github:

<https://github.com/RaghadAbdelhameed/ARC.git>