CARPATHIAN J. MATH.
Volume **42** (2026), No. 3,
Pages 661-678

Online version at https://www.carpathian.cunbm.utcluj.ro/
ISSN 1584-2851 (Print edition); ISSN 1843-4401 (Electronic)
DOI: https://doi.org/10.37193/CJM.2026.03.12

Double Inertial Shrinking Projection Algorithm for Fixed Point Problems with Deep Learning Integration for Lumbar Spinal Stenosis Detection

PAPINWICH PAIMSANG[1], WONGTHAWAT LIAWRUNGRUEANG[2], JAKKAPHONG INPAN[3], PHISIT KATONGTUNG[3], KANOKWATT SHIANGJEN[3], JEN-CHIH YAO[4], PRASIT CHOLAMJIAK[1], SUTHEP SUANTAI[5], AND WATCHARAPORN CHOLAMJIAK[1]

ABSTRACT. In this paper, we introduce a new modified hybrid iterative algorithm that integrates the shrinking projection method with double inertial extrapolations for solving common fixed point problems of a countable family of quasi-nonexpansive mappings in real Hilbert spaces. By incorporating double inertial terms, our scheme enables more efficient utilization of historical information from the iterates and significantly enhances convergence behavior. Under appropriate conditions, we prove a strong convergence theorem for the proposed method. To demonstrate its practical utility, we applied our algorithm to an automated classification task for lumbar spinal stenosis using axial T2-weighted MRI. The pipeline integrates YOLO segmentation, SSIM-based filtering, data augmentation, VGG19 feature extraction, and classification via a regularized ELM. Our model achieved 96.41% test accuracy and a macro-average AUC of 0.97. Comparative experiments with several standard machine learning models, including XGBoost, Random Forest, LightGBM, and SVM, demonstrated that our approach not only yields higher accuracy but also exhibits superior generalization performance, with minimal overfitting. These findings highlight the advantages of integrating advanced optimization algorithms with deep learning for improving classification performance in challenging medical image analysis tasks.

1. INTRODUCTION

Throughout this work, let C be a nonempty, closed, and convex subset of a real Hilbert space $\mathcal{H}$, equipped with the standard inner product $\langle \cdot, \cdot \rangle$ and the induced norm $\| \cdot \|$. Consider a mapping $T : C \to C$ that is nonexpansive, meaning that for all $x, y \in C$, we have

$$\|Tx - Ty\| \leq \|x - y\|.$$

We define the fixed point set of T as $F(T) := \{x \in C : Tx = x\}$.

Fixed point problems involving nonexpansive mappings arise in various fields, such as convex feasibility problems, monotone variational inequalities, convex optimization, and image reconstruction. It is well known that the classical Picard iteration scheme may not always converge when applied to nonexpansive operators. A well-established alternative is the Mann iteration method, which generates a sequence $\{x_n\}$ by the rule

$$x_{n+1} = \alpha_n x_n + (1 - \alpha_n)Tx_n, \quad n \geq 0,$$

where $\{\alpha_n\} \subset (0, 1)$. It has been shown that under the condition

$$\sum_{n=1}^{\infty} \alpha_n(1 - \alpha_n) = +\infty,$$

the sequence $\{x_n\}$ converges weakly to a fixed point of T.

Received: 11.07.2025. In revised form: 11.11.2025. Accepted: 25.12.2025
2020 *Mathematics Subject Classification.* 47H10, 47J25, 65K10, 68T07.
Key words and phrases. *Fixed point iteration, Inertial extrapolation, Strong convergence, Automated classification, Lumbar Spinal Stenosis.*
Corresponding author: Watcharaporn Cholamjiak; watcharaporn.ch@up.ac.th

The Mann iteration method has served as a foundational framework for approximating fixed points of nonexpansive operators in Hilbert spaces. Over the years, this concept has been extended in numerous directions to accommodate a broader range of theoretical settings and applications. One notable development involves hybrid iterative schemes that combine the ideas of Picard and Mann iterations [35], which have been shown to accelerate convergence under certain conditions. Other advancements include enriched iterative processes and inertial-type methods [28, 5], both of which enhance convergence speed and robustness. Furthermore, the integration of these techniques with shrinking projection strategies [42, 41] has led to strong convergence results, even in more complex scenarios such as pointwise nonexpansive mappings.

Inertial techniques have proven to be effective in accelerating the convergence of iterative algorithms by incorporating momentum-like terms derived from physical dynamics. These methods help maintain useful directions from previous iterates, thereby reducing oscillations and improving convergence speed. The theoretical foundation of inertial methods was laid by Alvarez and Attouch [1], who interpreted optimization dynamics as a discretization of a damped inertial system. Since then, numerous inertial-based algorithms have been proposed, such as the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [2], inertial extragradient methods [14], and inertial Mann-type iterations [37]. These methods have demonstrated significant improvements in applications including signal recovery, image reconstruction, and machine learning.

In 2008, Takahashi et al. [39] proposed a hybrid method that is different from the earlier method developed by Nakajo and Takahashi [29]. This approach is now commonly referred to as the *shrinking projection method*. Motivated by their work, we consider the following modified hybrid iterative scheme, which has since become a fundamental tool in fixed point theory. The method generates a sequence $\{x_n\}$ according to the following recursive procedure: $x_0 \in C$, $C_1 = C$,

$$\begin{aligned} y_n &= \alpha_n x_n + (1-\alpha_n)Tx_n, \\ C_{n+1} &= \{v \in C_n : \|y_n - v\| \le \|x_n - v\|\}, \\ x_{n+1} &= P_{C_{n+1}} x_0, \end{aligned} \tag{1.1}$$

for each $n \in \mathbb{N}$, where $0 \le \alpha_n \le \alpha < 1$ and $P_{C_{n+1}} x_0$ denotes the metric projection of x_0 onto the closed convex subset C_{n+1}. The underlying principle of this approach is to iteratively refine the constraint set C_n by selecting elements based on the inequality $\|y_n - v\| \le \|x_n - v\|$, which ensures that the sequence is progressively directed toward the fixed point set of the mapping T.

The inertial Mann-type algorithm, introduced by Maingé [28], has been recognized as an effective extension of classical fixed point schemes, particularly for addressing slow convergence. By introducing an extrapolated iterate using the momentum of previous steps, the algorithm improves efficiency while retaining the simplicity of projection-type updates. In Maingé's formulation, the algorithm employs a sequence of self-mappings $\{T_n\}\colon \mathcal{H} \to \mathcal{H}$, and is defined as follows:

$$\begin{aligned} y_n &= x_n + \theta_n(x_n - x_{n-1}), \\ x_{n+1} &= (1-w_n)y_n + w_n T_n y_n, \quad n \ge 1, \end{aligned} \tag{1.2}$$

where $\theta_n \in [0,1]$ is the inertial parameter and $w_n \in (0,2)$ is a relaxation factor. The mappings T_n are assumed to be nonexpansive with a common fixed point set $S = \bigcap_{n\ge 0} \mathrm{Fix}(T_n) \neq \emptyset$.

The idea of incorporating double inertial into iterative fixed point algorithms has gained increasing attention due to its potential to improve convergence behavior. Unlike standard inertial schemes, which rely on a single extrapolation term, double inertial methods make use of additional historical information to enhance stability and accelerate convergence. This has been supported by several studies, including those of Combettes and Glaudin [9], Dong et al. [15], Iyiola and Shehu [21], Li et al. [27], and Polyak [30], which demonstrated that multi-step inertial terms such as the two-step formulation can significantly outperform single-step variants, both in theory and in practice. These findings motivate further investigation of two-step inertial strategies in various algorithmic frameworks.

To enhance the efficiency of inertial schemes, various researchers have explored double extensions. One such development is the two-step inertial proximal point algorithm introduced by Iyiola and Shehu [21], which is defined as follows:

$$\begin{aligned} x_{n+1} &= J_\lambda^A(y_n), \\ y_{n+1} &= x_{n+1} + \theta(x_{n+1} - x_n) + \delta(x_n - x_{n-1}), \end{aligned}$$

where $\delta, \theta \in [0, 1)$ are the inertial parameters, and $J_\lambda^A = (I + \lambda A)^{-1}$ denotes the resolvent of a maximal monotone operator A with $\lambda > 0$.

In this paper, we propose a novel fixed point iterative method that combines the shrinking projection technique with two-step inertial extrapolation. We establish strong convergence results for the proposed algorithm in the context of finding a common fixed point of a family of nonexpansive mappings. In addition, we support the theoretical findings with numerical experiments conducted on the lumbar central spinal stenosis grade classification dataset.

2. Preliminaries

In this section, we provide some fundamental theorem and lemmas that will be used throughout the paper.

We will denote the set of fixed points of the operator T: $C \to C$ by $F(T) = \{x \in C : Tx = x\}$. For the sequence $\{x_n\}$ to x in C, the strong convergence and the weak convergence are denoted by $x_n \to x$ and $x_n \rightharpoonup x$, respectively. An operator T on C is nonexpansive if, for each $x, y \in C$,

$$\|Tx - Ty\| \leq \|x - y\|.$$

T is said to be quasi-nonexpansive if $F(T) \neq \emptyset$, and for any $x \in C$ and $p \in F(T)$,

$$\|Tx - p\| \leq \|x - p\|.$$

Let $\mathcal{H}$ be a Hilbert space and let $C \subset \mathcal{H}$. Let $\{T_n\}$ be a sequence and τ a family of self-mappings on C. Assume that the sets of fixed points $F(\tau)$ and $\bigcap_{n=1}^{\infty} F(T_n)$ are nonempty, where $F(T_n)$ denotes the set of fixed points of T_n, and $F(\tau)$ denotes the set of common fixed points of all mappings in τ. The sequence $\{T_n\}$ is said to satisfy the NST_w-*condition* with respect to τ if, for every bounded sequence $\{x_n\} \subset C$, there exists a subsequence $\{x_{n_j}\} \subset \{x_n\}$ such that

$$\lim_{n\to\infty} \|x_n - T_n x_n\| = 0 \quad \Rightarrow \quad \lim_{j\to\infty} \|x_{n_j} - T x_{n_j}\| = 0, \quad \forall T \in \tau.$$

The notion of the NST_w-*condition* was introduced in [8].

Lemma 2.1 ([38])**.** *Let $\mathcal{H}$ be a real Hilbert space. Then, for all $x, y \in \mathcal{H}$, the following statements hold:*

(1) $\|x - y\|^2 = \|x\|^2 - \|y\|^2 - 2\langle x - y, y\rangle$;

(2) $\|x+y\|^2 \leq \|x\|^2 + 2\langle y, x+y\rangle$;
(3) $\|tx+(1-t)y\|^2 = t\|x\|^2 + (1-t)\|y\|^2 - t(1-t)\|x-y\|^2$, *for all* $t \in [0,1]$.

Lemma 2.2 ([24]). *Let $C \subset \mathcal{H}$ be a nonempty, closed, and convex subset of a real Hilbert space $\mathcal{H}$. For any $x, y, z \in \mathcal{H}$ and $a \in \mathbb{R}$, the set*

$$\left\{v \in C : \|y-v\|^2 \leq \|x-v\|^2 + \langle z, v\rangle + a\right\}$$

is convex and closed.

Lemma 2.3 ([29]). *Let $C \subset \mathcal{H}$ be a nonempty, closed, and convex subset of a real Hilbert space $\mathcal{H}$, and let $P_C : \mathcal{H} \to C$ denote the metric projection from $\mathcal{H}$ onto C. Then, for all $x \in \mathcal{H}$ and $y \in C$, the following inequality holds:*

$$\|y - P_C x\|^2 + \|x - P_C x\|^2 \leq \|x-y\|^2.$$

3. Algorithm and convergence analysis

Let C be a nonempty closed convex subset of a real Hilbert space $\mathcal{H}$. Let $\{T_n\}$ and τ be families of quasi-nonexpansive mappings into C such that $\{T_n\}$ satisfies the NST_w-*condition* with respect to τ.

Algorithm 3.1. *Double Inertial Shrinking Projection Algorithm*

Initialization : *Let $x_0, y_0, y_{-1} \in C$ be arbitrary, $\theta_n, \delta_n \in (-\infty, \infty)$, $\alpha_n \in (0,1)$ and set $C_0 = C$.*
Step 1: *Set $n = 0$, compute*

$$y_{n+1} = (1-\alpha_n)x_n + \alpha_n T_n x_n.$$

Step 2:

$$z_{n+1} = y_{n+1} + \theta_n(y_{n+1} - y_n) + \delta_n(y_n - y_{n-1}).$$

Step 3:

$$\nabla_n = \theta_n(y_{n+1} - y_n) + \delta_n(y_n - y_{n-1}).$$

Step 4:

$$C_{n+1} = \left\{u \in C_n : \|z_{n+1} - u\|^2 \leq \|x_n - u\|^2 + 2\langle y_{n+1} - u, \nabla_n\rangle + \|\nabla_n\|^2\right\}.$$

Step 5:

$$x_{n+1} = P_{C_{n+1}} x_0.$$

Then, $n = n+1$ and update in ***Step 1.***

Theorem 3.1. *Let $\mathcal{H}$ be a real Hilbert space and C be nonempty closed convex subset of $\mathcal{H}$. Let $\{T_n\}$ and τ be families of quasi-nonexpansive mappings such that $\{T_n\}$ satisfies NST_w-condition with respect to τ. Suppose that $F(\tau)$, $\bigcap_{n=0}^{\infty} F(T_n)$ are nonempty sets. Let $\{x_n\}$ be defined by Algorithm 3.1. Assume that the following conditions are satisfied:*

(i) $\lim_{n\to\infty} |\theta_n|\|y_{n+1} - y_n\| = 0$;
(ii) $\lim_{n\to\infty} |\delta_n|\|y_n - y_{n-1}\| = 0$;
(iii) $0 < \liminf_{n\to\infty} \alpha_n \leq \limsup_{n\to\infty} \alpha_n < 1$;
(iv) $I - T$ *is demiclosed at 0 for all* $T \in \tau$.

Then, $\{x_n\}$ converges strongly to a point in $F(\tau)$.

Proof. We split the proof into four steps.

Step 1. Show that $P_{C_{n+1}}x_0$ is well-defined for each $x_0 \in C$.

From the definition of C_{n+1}, from Lemma 2.2, C_{n+1} is closed and convex for each $n \geq 1$. For each $n \in \mathbb{N}$, let $q \in \bigcap_{n=1}^{\infty} F(T_n)$. Since T_n is quasi-nonexpansive, we have

$$\begin{aligned}\|y_{n+1} - q\|^2 &= \|(1-\alpha_n)x_n + \alpha_n T_n x_n - q\|^2 \\ &= (1-\alpha_n)\|x_n - q\|^2 + \alpha_n\|T_n x_n - q\|^2 - \alpha_n(1-\alpha_n)\|T_n x_n - x_n\|^2 \\ &\leq (1-\alpha_n)\|x_n - q\|^2 + \alpha_n\|x_n - q\|^2 - \alpha_n(1-\alpha_n)\|T_n x_n - x_n\|^2 \\ &= \|x_n - q\|^2 - \alpha_n(1-\alpha_n)\|T_n x_n - x_n\|^2, \end{aligned} \tag{3.3}$$

and

$$\begin{aligned}\|z_{n+1} - q\|^2 &= \|y_{n+1} - q + \theta_n(y_{n+1} - y_n) + \delta_n(y_n - y_{n-1})\|^2 \\ &= \|y_{n+1} - q\|^2 + 2\langle \theta_n(y_{n+1} - y_n) + \delta_n(y_n - y_{n-1}), y_{n+1} - q\rangle \\ &\quad + \|\theta_n(y_{n+1} - y_n) + \delta_n(y_n - y_{n-1})\|^2 \\ &\leq \|x_n - q\|^2 + 2\langle \nabla_n, y_{n+1} - q\rangle + \|\nabla_n\|^2. \end{aligned}$$

Therefore, we have $q \in C_{n+1}$, and thus, $\bigcap_{n=0}^{\infty} F(T_n) \subset C_{n+1}$. Therefore, $P_{C_{n+1}}x_0$ is well defined.

Step 2: Prove that $\lim_{n\to\infty} \|x_n - x_0\|$ exists. Since $\bigcap_{n=0}^{\infty} F(T_n)$ is a nonempty, closed, and convex subset of $\mathcal{H}$, there exists a unique point $p \in \bigcap_{n=0}^{\infty} F(T_n)$ such that $p = P_{\bigcap_{n=0}^{\infty} F(T_n)}x_0$. From the fact that $x_n = P_{C_n}x_0$ and $x_{n+1} \in C_{n+1} \subset C_n$ for all $n \in \mathbb{N}$, it follows that

$$\|x_n - x_0\| \leq \|x_{n+1} - x_0\|, \quad \forall n \in 0.$$

On the other hand, since $\bigcap_{n=0}^{\infty} F(T_n) \subset C_n$ for all $n \geq 0$, we obtain

$$\|x_n - x_0\| \leq \|p - x_0\|, \quad \forall n \geq 0.$$

Hence, the sequence $\{x_n\}$ is nondecreasing and bounded. Therefore, $\lim_{n\to\infty} \|x_n - x_0\|$ exists.

Step 3: Show that $x_n \to q \in C$ as $n \to \infty$. For $m > n$, by the definition of C_n, we see that $x_m = P_{C_m}x_1 \in C_m \subset C_n$. From Lemma 2.3, we have

$$\|x_m - x_n\|^2 \leq \|x_m - x_0\|^2 - \|x_n - x_0\|^2.$$

From Step 3, we obtain that $\{x_n\}$ is a Cauchy sequence. Hence, there exists $q \in \mathcal{H}$ such that $x_n \to q$ as $n \to \infty$. In particular, we have

$$\lim_{n\to\infty} \|x_{n+1} - x_n\| = 0. \tag{3.4}$$

Step 4: Show that $q \in \bigcap_{n=0}^{\infty} F(T_n)$. From $x_{n+1} \in C$, from (3.4) and conditions (i), (ii), we obtain

$$\begin{aligned}\|z_{n+1} - x_{n+1}\|^2 &\leq \|x_n - x_{n+1}\|^2 + 2\langle y_{n+1} - x_{n+1}, \nabla_n\rangle + 2\|\nabla_n\|^2 \\ \|z_{n+1} - x_{n+1}\| &\leq \sqrt{\|x_n - x_{n+1}\|^2 + 2\langle y_{n+1} - x_{n+1}, \nabla_n\rangle + 2\|\nabla_n\|^2} \\ &\to 0. \end{aligned} \tag{3.5}$$

It follows from (3.4) and (3.5) that

$$\|z_{n+1} - x_n\| \leq \|z_{n+1} - x_{n+1}\| + \|x_{n+1} + x_n\| \to 0 \tag{3.6}$$

From (3.3), we have

$$\begin{aligned}\|z_{n+1}-q\|^2 &= \|y_{n+1}+\theta_n(y_{n+1}-y_n)+\delta_n(y_n-y_{n-1})-q\|^2\\ &\leq \|y_{n+1}-q\|^2+2\left\langle\theta_n(y_{n+1}-y_n)+\delta_n(y_n-y_{n-1}),\, z_{n+1}-q\right\rangle\\ &\leq \|x_n-q\|^2-\alpha_n(1-\alpha_n)\|T_nx_n-x_n\|^2+2\langle\theta_n(y_{n+1}-y_n)\\ &\quad+\delta_n(y_n-y_{n-1}),\, z_{n+1}-q\rangle.\end{aligned}$$

So we have

$$\begin{aligned}\|T_nx_n-x_n\|^2 \leq &\frac{1}{\alpha_n(1-\alpha_n)}(\|z_{n+1}-x_n\|\left(\|z_{n+1}-q\|+\|x_n-q\|\right)+2\langle\theta_n(y_{n+1}-y_n)\\ &+\delta_n(y_n-y_{n-1}),\, z_{n+1}-q\rangle).\end{aligned}$$

It follows from above, the Assumptions (i)- (iii), and (3.6) that

$$\lim_{n\to\infty}\|T_nx_n-x_n\|=0.$$

Since $\{T_n\}$ satisfies the NST_w-*condition* with respect to τ, there exists a subsequence $\{x_{n_j}\}$ of $\{x_n\}$ such that

$$\lim_{j\to\infty}\|x_{n_j}-Tx_{n_j}\|=0, \quad \text{for all } T\in\tau. \tag{3.7}$$

By assumption (iv), $I-T$ is demiclosed at zero for all $T\in\tau$, and using (3.7) and from Step 3, $x_n\to q$, it follows that $q\in F(\tau)$. This completes the proof. □

4. Application to Lumbar Spinal Stenosis Classification

In this study, we developed a classification framework to automatically distinguish between different morphological grades of lumbar spinal stenosis using axial T2-weighted MRI images. The entire workflow consists of preprocessing, segmentation, feature extraction, and final classification using Extreme Learning Machine model. All experimental procedures were implemented using Python with the support of libraries such as NumPy, scikit-learn, and TensorFlow. An overview of the proposed pipeline is illustrated in Figure 1, which summarizes the key stages from input image acquisition to final grade prediction.

4.1. **YOLO-Based Segmentation.**

YOLO (You Only Look Once) is a family of object detection models that revolutionized real-time detection by unifying classification and localization into a single network pass. The original YOLO model (YOLOv1) introduced by Redmon et al. [32] emphasized speed and simplicity, making it one of the first methods capable of real-time object detection on consumer-grade GPUs.

These advances make YOLOv11 particularly suitable for medical imaging applications, such as our lumbar spinal canal segmentation, where both speed and precision are essential.

4.2. **Data Augmentation Techniques.**

Data augmentation is a fundamental strategy in deep learning, particularly for image classification. One of the earliest uses appeared in LeNet-5 by LeCun et al., where geometric warping was employed to improve handwritten digit recognition [26]. Later, augmentation became an effective remedy for class imbalance through oversampling methods. Chawla et al. introduced SMOTE [6], followed by Borderline-SMOTE [19], which generate synthetic samples by interpolating minority class instances using nearest neighbors. These methods were originally designed for tabular data but inspired similar strategies in image domains.

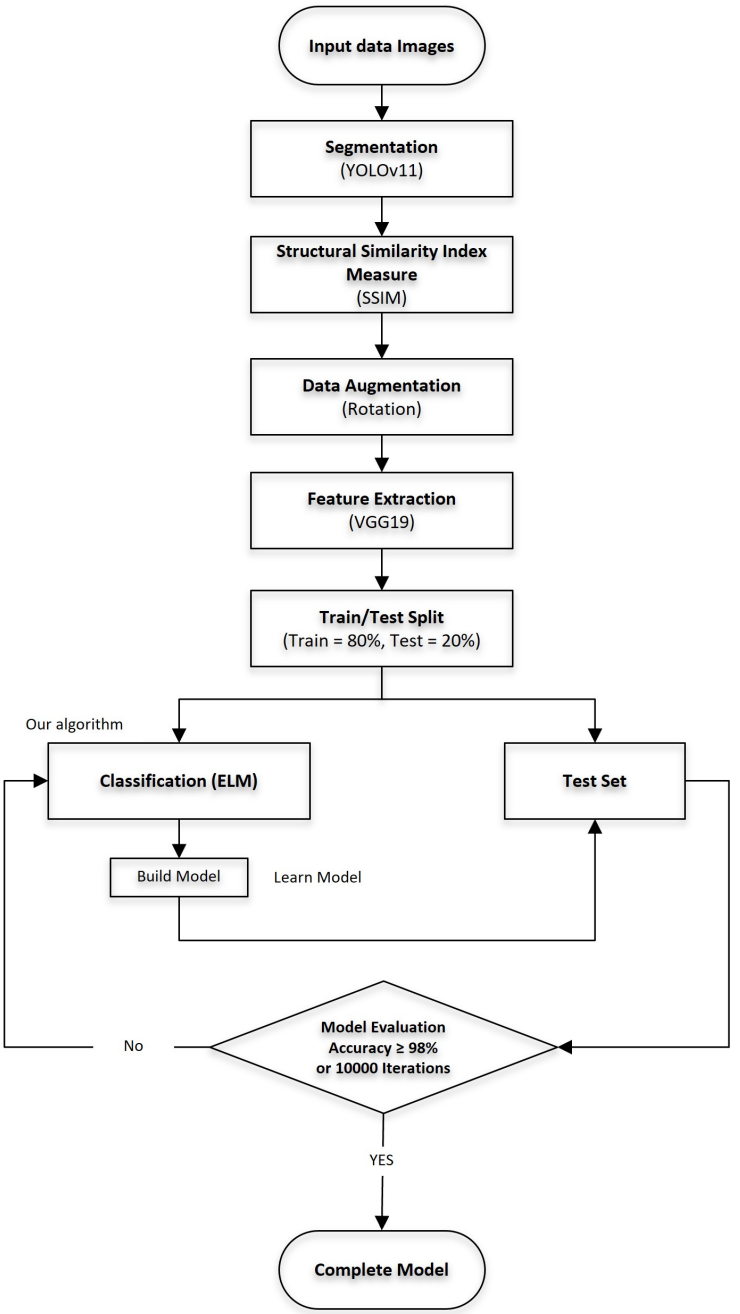


FIGURE 1. Work flow diagram

A significant breakthrough came with AlexNet [25], which applied random crops, horizontal flips, and PCA-based color shifts to expand the training set by over 2,000 times, effectively reducing overfitting. Since then, advanced techniques have emerged, including

GAN-based synthetic image generation [18], Neural Style Transfer [16], and AutoAugment [12], which use meta-learning to search for optimal augmentation policies. These advances have made data augmentation a cornerstone in modern CNN pipelines.

4.3. **CNN and Feature Extraction using VGG19.** Convolutional Neural Networks (CNNs) are widely used for feature extraction due to their ability to learn spatial hierarchies of patterns in images. VGG19, introduced by Simonyan and Zisserman [36], is a deep CNN composed of 19 layers with uniform 3×3 convolutional filters and max-pooling. Pre-trained on ImageNet, VGG19 can transfer learned features to medical imaging tasks with limited data.

4.4. **Structural Similarity Index Measure (SSIM).** The Structural Similarity Index Measure (SSIM) is a perceptual metric for quantifying similarity between two images. It evaluates luminance, contrast, and structure, and is defined as:

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \tag{4.8}$$

where μ_x, μ_y are means, σ_x^2, σ_y^2 variances, and σ_{xy} the covariance of images x and y. SSIM values range from 0 (dissimilar) to 1 (identical). It is widely used in medical image quality assessment due to its alignment with human visual perception.

4.5. **Proposed Framework Using ELM.** The proposed classification pipeline consists of the following components:

Region of Interest (ROI) Segmentation: We employ YOLOv5 to automatically segment the lumbar spinal canal from axial T2-weighted MRI slices. This ensures that the model focuses on clinically relevant regions with high localization precision.

Image Quality Filtering using SSIM: The Structural SSIM is used to compare each cropped image against a representative reference image from its class. Samples with low SSIM values are discarded to ensure high structural consistency in the training dataset.

Data Augmentation: To mitigate class imbalance and enhance model robustness, we apply random in-plane rotations within the range of $\pm 15^\circ$. This introduces anatomical variability while preserving core structural patterns.

Feature Extraction using VGG19: Augmented images are passed through a pre-trained VGG19 model (trained on ImageNet), and the convolutional and pooling layers are retained to extract deep hierarchical features suited for classification.

Classification with Extreme Learning Machine (ELM): The resulting feature vectors are fed into an ELM model to perform final classification into the seven morphological grades (A1–D).

Lumbar spinal stenosis is a degenerative disease caused by the narrowing of the spinal canal, leading to compression of the spinal cord or nerve roots. Clinically, this manifests as lower back pain, neurogenic claudication, and lower limb weakness, particularly in elderly individuals. Accurate assessment of stenosis severity is essential for treatment planning, and axial T2-weighted Magnetic Resonance Imaging (MRI) is widely used to evaluate the dural sac morphology. The classification system proposed by Schizas et al. [34] categorizes the dural sac into seven morphological grades, ranging from A1 to D.

To automate this classification, we adopt the framework of single-layer feedforward neural networks (SLFNs) using the ELM approach, as introduced by Huang et al. [20]. The training dataset is denoted by $\mathcal{U} := \{(d_n, t_n) : d_n \in \mathbb{R}^k, t_n \in \mathbb{R}^m, n = 1, 2, \ldots, N\}$,

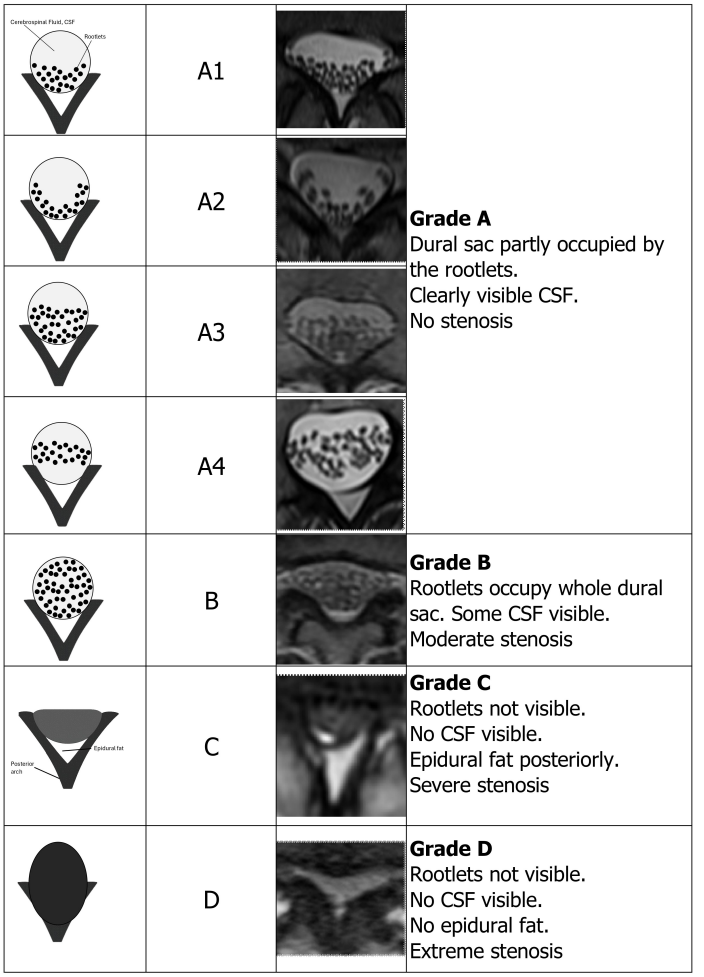


FIGURE 2. Graphic and magnetic resonance images showing the morphological classification of severity of lumbar spinal stenosis of Schizas et al. [34]

where d_n is the is input training data, t_n is the corresponding target label, and N is the number of training samples. The output function for an ELM is given by:

$$O_k = \sum_{j=1}^{M} \beta_j A(w_j d_k + b_j), \tag{4.9}$$

where M is the number of hidden neurons, A is the activation function, and w_j, b_j are randomly generated weights and biases, while β_j are the output weights to be determined. The hidden layer output matrix P is then constructed as:

$$P = \begin{bmatrix} A(w_1 d_1 + b_1) & \cdots & A(w_M d_1 + b_M) \\ \vdots & \ddots & \vdots \\ A(w_1 d_N + b_1) & \cdots & A(w_M d_N + b_M) \end{bmatrix}. \tag{4.10}$$

The optimal output weight vector is defined as $\beta = [\beta_1^T, \ldots, \beta_M^T]^T$, where each $\beta_j \in \mathbb{R}^m$ corresponds to the output associated with the j-th hidden neuron. The target output matrix is denoted as $Z = [t_1^T, \ldots, t_N^T]^T \in \mathbb{R}^{N\times m}$, where each $t_n \in \mathbb{R}^m$ represents the desired label corresponding to the input sample d_n.

Finding the optimal output-weight vector $\beta \in \mathbb{R}^{M\times m}$ can be written as the minimization of the least-squares objective

$$\min_{\beta\in\mathbb{R}^M} f(\beta) \;=\; \min_{\beta\in\mathbb{R}^M} \left(\frac{1}{2} \|P\beta - Z\|_2^2 \right), \tag{4.11}$$

where $P \in \mathbb{R}^{N\times M}$ is the hidden-layer output matrix and $Z \in \mathbb{R}^{N\times m}$ is the matrix of target outputs.

By the first-order theory of Bertsekas et al. [3] and the fixed-point framework of Rockafellar [33], the minimizer β^* under the chain of equivalent conditions.

$$\beta^* \in \arg\min_{\beta\in\mathbb{R}^M} f(\beta) \iff \nabla f(\beta^*) = 0 \iff \beta^* = \big(I - \lambda\nabla f\big)(\beta^*), \quad \forall\, \lambda > 0.$$

If $P^T P$ is invertible, the closed-form solution is

$$\beta^* \;=\; (P^T P)^{-1} P^T Z.$$

Otherwise, any gradient step size $\lambda \in \big(0, 2/L\big]$, where $L = \|P^T P\|^2$ is the Lipschitz constant of ∇f, $T(\cdot) = \big(I - \lambda\nabla f\big)(\cdot)$ is quasi-nonexpansive (Lemma 5.1.18 [8]).

Accordingly, our proposed algorithm 3.1 can be employed to solve problem (4.11) by applying the operator

$$T_n x_n = \big(I - \lambda\nabla f\big)(x_n),$$

with the choice $\lambda = \frac{1.99}{\|P^T P\|^2}$ keeps the step size safely within the permitted range.

To comprehensively evaluate the effectiveness of the proposed classification framework, we employed widely accepted performance metrics, including precision, recall, F1-score, and accuracy. These metrics offer insights into the model's ability to correctly classify instances, handle imbalanced data, and maintain consistent prediction quality across different classes. The formal definitions are as follows:

$$\begin{aligned}
\text{Precision}(\%) &= \frac{\text{tp}}{\text{tp} + \text{fp}} \times 100, \\
\text{Recall}(\%) &= \frac{\text{tp}}{\text{tp} + \text{fn}} \times 100, \\
\text{F1-score}(\%) &= \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \\
\text{Accuracy}(\%) &= \frac{\text{tp} + \text{tn}}{\text{tp} + \text{fp} + \text{tn} + \text{fn}} \times 100,
\end{aligned}$$

where tp, tn, fp, fn denote true positives, true negatives, false positives, and false negatives, respectively.

Additionally, for multi-class classification, One-vs-Rest Receiver Operating Characteristic (One-vs-Rest ROC) curves were used to evaluate the model's discriminative ability across each class. In this approach, the ROC curve is computed by treating one class as positive and the rest as negative for each iteration. The macro-average ROC AUC is then obtained by averaging the Area Under the Curve (AUC) values of each class equally.

The true positive rate (TPR) and false positive rate (FPR) are defined as:

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \qquad \text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}},$$

and the AUC for a class is computed as:

$$\text{AUC}_c = \int_0^1 \text{TPR}_c(\text{FPR}_c)\, d\text{FPR}_c,$$

where the subscript c refers to a specific class. The final macro-average AUC is:

$$\text{AUC}_{\text{macro}} = \frac{1}{C}\sum_{c=1}^{C} \text{AUC}_c,$$

where C is the number of classes.

We also computed the multi-class cross-entropy loss to evaluate prediction confidence across all classes:

$$\text{Loss} = -\sum_{j=1}^{N} o_j \log \hat{o}_j,$$

where $\hat{o}_j$ is the j-th scalar value in the model output, o_j is the corresponding target value, and N is the number of scalar values.

The ELM model is solved by setting the gradient terms $f(\beta) = \frac{1}{2}\|P\beta - \mathbf{Z}\|_2^2$ and $g(\beta) = \partial(\lambda\|\beta\|_1)$. To support our main theorem, the parameters θ_n and δ_n are defined as follows:

$$\theta_n = \begin{cases} \min\left(\frac{1}{n^2\|y_{n+1}-y_n\|+1}, \theta_{\max}\right), & \text{if } y_{n+1} \neq y_n,\ n > N \\ \theta & \text{otherwise} \end{cases} \tag{4.12}$$

and

$$\delta_n = \begin{cases} \min\left(\frac{1}{n^2\|y_{n+1}-y_n\|+1}, \delta_{\max}\right), & \text{if } y_{n+1} \neq y_n,\ n > N \\ \delta & \text{otherwise.} \end{cases} \tag{4.13}$$

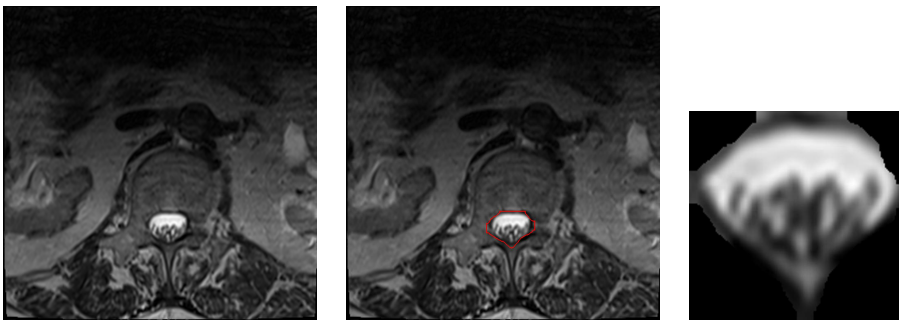
The parameters of our Algorithm are defined by $\alpha_n = 0.9$, $\lambda = \dfrac{1.99}{\|P^T P\|^2}$, and θ_n, δ_n are defined by the same as (4.12) and (4.13) such that $\theta = -7.14$, $\theta_{max} = 10$, $\delta = 7.33$ and $\delta_{max} = 10$, when $M = 1000$.

The input MRI data used in this study were in the form of axial T2-weighted slices from the RSNA 2024 dataset, with the dural sac manually delineated in clinical ground truth. Before classification, we applied YOLOv11 segmentation to isolate the lumbar spine canal from MRI slices. This focuses learning on clinically relevant regions, removes extraneous background, and improves model efficiency. The segmented regions were resized to 512×512 (Figure 3).

To address the issue of class imbalance, particularly in underrepresented morphological grades, we applied data augmentation techniques involving random in-plane rotations within $\pm 15^\circ$. This strategy increased the diversity of the training data and improved the model's ability to generalize to unseen cases. Before augmentation, the segmented images were filtered using SSIM [40] by comparing each image with the best representative image of its class. Images with low SSIM scores, indicating significant structural dissimilarity, were considered erroneous and excluded from the dataset.

After this process, the dataset consisted of a total of 15,371 images. To prepare the data for training and evaluation, we randomly split the dataset into 80% for training and 20% for testing. The number of samples in each class after the split is shown in Table 1.

To extract meaningful hierarchical features from the input MRI slices, we employed the VGG19 model as the convolutional backbone The model was pre-trained on the ImageNet dataset, enabling it to provide rich and transferable representations, even for tasks outside the original training domain. This deep architecture allows effective encoding



(a) Original MRI Slice (b) Segmentation MRI Slice (c) Cropped ROI

FIGURE 3. Example of segmentation process: original axial MRI slice (a), YOlO segmentation (b) and cropped lumbar spine region (c).

TABLE 1. Class distribution of training and testing samples across all seven morphological grades (A1–D). The dataset was partitioned into 80% for training and 20% for testing.

Grade	Train Samples	Test Samples	Total
A1	1887	471	2358
A2	1811	466	2277
A3	1821	444	2265
A4	1832	499	2331
B	1740	466	2206
C	1591	341	1932
D	1614	388	2002
Total	**12,296**	**3,075**	**15,371**

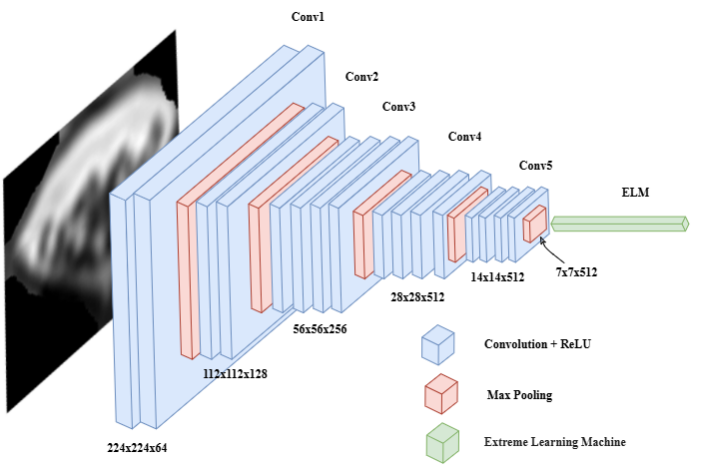


FIGURE 4. The modified VGG19 model

of both low-level texture information and high-level semantic patterns, making it particularly suitable for analyzing complex medical images. Integrating VGG19 into our pipeline

improved feature expressiveness and enhanced overall classification performance (Figure 4).

TABLE 2. Comparison of training time and classification performance across three algorithms.

Algorithm	Time (sec)	Precision	Recall	F1-score	Accuracy
Algorithm 3.1	9,873.48	87.91%	88.12%	87.79%	96.41%
Algorithm (1.1)	9,729.59	85.55%	87.81%	86.64%	94.35%
Algorithm (1.2)	9,116.01	78.73%	80.94%	79.79%	89.76%

Table 2 compares the training time and classification performance of three algorithms. Although Algorithm 1.1 required the least training time (9,729.59 seconds), it yielded the lowest classification accuracy (94.35%). In contrast, Algorithm 3.1 achieved the highest accuracy (96.41%) and F1-score (87.79%), making it the most effective in terms of predictive performance. These results suggest a trade-off between speed and classification quality, with Algorithm 3.1 offering better accuracy at a slightly higher computational cost.

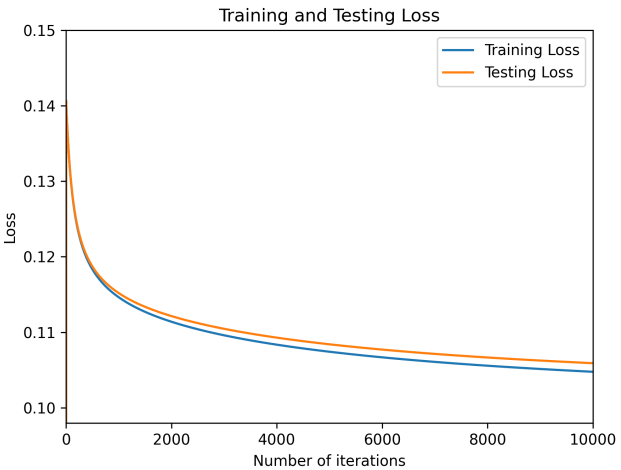


FIGURE 5. Training and testing loss across 10,000 iterations.

Figure 5 illustrates the evolution of training and testing loss over 10,000 iterations. The model exhibits a sharp decline in both losses during the early stages, reflecting efficient learning and rapid convergence. As iterations progress, the loss curves stabilize with minimal fluctuation, and the gap between them remains narrow throughout. The final training and testing loss values were 0.1048 and 0.1059, respectively, indicating that the model achieved low classification error while maintaining strong generalization without significant overfitting.

Figure 6 illustrates the accuracy trajectories of the training and testing sets over 10,000 iterations. The training accuracy shows a steady and smooth increase, indicating that the model effectively learns from the training data. Simultaneously, the test accuracy follows a nearly parallel trend, suggesting minimal overfitting. The small gap between the two curves throughout training reflects strong generalization capability. This performance

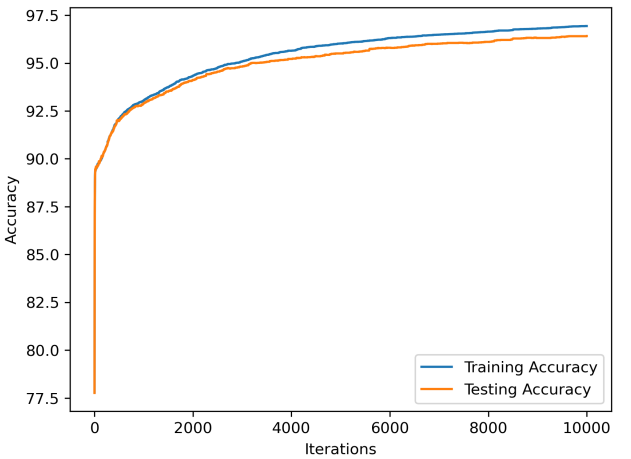


FIGURE 6. Training and testing accuracy across 10,000 iterations.

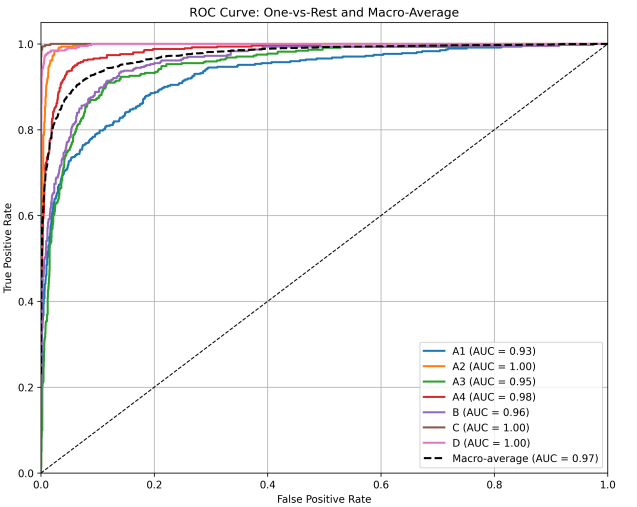


FIGURE 7. One-vs-Rest ROC curves and macro-average for each grade (A1–D).

highlights the effectiveness of our algorithm in optimizing classification without sacrificing robustness to unseen data.

To further assess the discriminative ability of the model across classes, we plotted the One-vs-Rest ROC curves for each morphological grade, as shown in Figure 7. AUC values ranged from 0.93 to 1.00, with a macro-average AUC of 0.97, suggesting excellent classification performance.

Finally, the confusion matrix in Figure 8 shows the model's class-wise predictions. Strong diagonal dominance indicates accurate predictions for most classes. However, some degree of misclassification is observed between morphologically adjacent classes, which aligns with clinical subjectivity in grading.

These consistent metrics across both training and testing sets confirm that the model is not only accurate but also well-calibrated and generalizable. The strong agreement

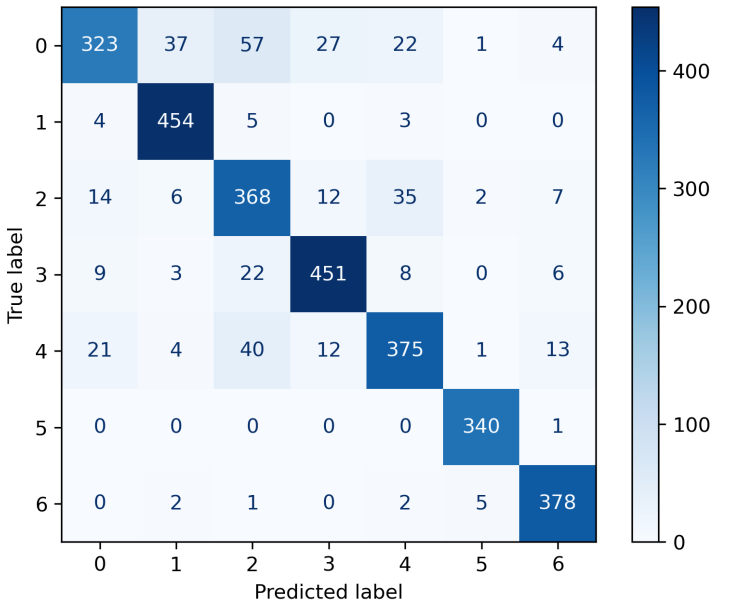


FIGURE 8. Confusion matrix of the model predictions across 7 classes (A1–D).

between the training and testing results further validates the effectiveness of the applied pre-processing techniques, segmentation strategy, and model architecture.

To further demonstrate the effectiveness of the proposed algorithm, we conducted a comparative analysis with widely used machine learning models on the same dataset and under identical data preprocessing conditions. The comparison results, in terms of training and testing accuracy, are presented in Table 3.

TABLE 3. Comparison of Classification Performance among Different Algorithms

Model	**Train Accuracy (%)**	**Test Accuracy (%)**
XGBoost [7]	99.85	95.77
Extra Trees [17]	99.85	95.57
LightGBM [23]	99.85	95.57
Random Forest [4]	99.85	95.18
Logistic Regression [13]	99.53	94.89
SVM [10]	95.91	93.26
K-Nearest Neighbors [11]	93.97	90.27
Decision Tree [31]	99.85	86.89
Naive Bayes [22]	72.19	69.04
Proposed Algorithm (Algorithm 3.1)	**96.94**	**96.41**

As shown in Table 3, the proposed algorithm achieved a Test Accuracy of **96.41%**, which is superior to all baseline models. Importantly, both Train and Test Accuracy of the proposed method are identical, indicating excellent generalization ability without overfitting. In contrast, most conventional models, such as XGBoost, Extra Trees, and Random Forest, exhibit extremely high training performance but show a noticeable performance drop on

testing data, reflecting overfitting tendencies. These findings confirm that the proposed algorithm offers a more robust and reliable solution for the classification task.

5. Conclusion

In this study, we investigated a fixed-point problem using the proposed Double Inertial Shrinking Projection Algorithm and established its strong convergence under appropriate conditions. To demonstrate its practical utility, we applied the algorithm to a medical image classification task: automated grading of lumbar spinal stenosis severity based on axial T2-weighted MRI scans. Deep convolutional features were extracted using the pre-trained VGG19 model, and classification was performed through an ELM, in which the output weights were optimized using our proposed algorithm formulated as an ℓ_1-regularized least-squares problem. YOLO-based segmentation was applied prior to classification to isolate clinically relevant spinal regions, and data augmentation techniques were employed to alleviate class imbalance. The resulting model achieved a test accuracy of 96.41% and a macro-average AUC of 0.97.

Furthermore, we conducted a comparative evaluation of the proposed model against several widely used machine learning algorithms, including XGBoost, Random Forest, LightGBM, and Support Vector Machines, under identical dataset and preprocessing conditions. The proposed model not only outperformed all baseline methods in terms of test accuracy, but also exhibited superior generalization capability without overfitting, as evidenced by the consistent train and test accuracy. These results highlight the advantages of integrating advanced optimization techniques with deep feature representations for solving complex medical classification problems and demonstrate the competitiveness of our proposed approach compared to conventional machine learning models.

Acknowledgments

This research was funded by Chiangmai University and University of Phayao and Thailand Science Research and Innovation Fund (Fundamental Fund 2026, Grant No. 2251/2568)

Data Availability Statement. The MRI data used in this study are publicly available from the RSNA 2024 Lumbar Spine Degenerative Classification Challenge. Interested readers can access the dataset at:
https://www.kaggle.com/competitions/rsna-2024-lumbar-spine-degenerative-classification

References

[1] Alvarez, F.; Attouch, H. An inertial proximal method for maximal monotone operators via discretization of a nonlinear oscillator with damping. *Set-Valued Anal.* **9** (2001), 3–11.
[2] Beck, A.; Teboulle, M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.* **2** (2009), 183–202.
[3] Bertsekas, D.; Nedić, A.; Ozdaglar, A. *Convex Analysis and Optimization*. Athena Scientific, Belmont, MA, 2003.
[4] Breiman, L. Random forests. *Machine Learning* **45** (2001), no. 1, 5–32.
[5] Boţ, R. I.; Csetnek, E. R. Inertial Douglas–Rachford splitting for monotone inclusion problems. *Appl. Math. Comput.* **256** (2015), 472–487.
[6] Chawla, N. V.; Bowyer, K. W.; Hall, L. O.; Kegelmeyer, W. P. SMOTE: Synthetic Minority Over-Sampling Technique. *J. Artif. Intell. Res.* **16** (2002), 321–357.
[7] Chen, T.; Guestrin, C. XGBoost: A scalable tree boosting system. In *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, 2016, 785–794.
[8] Cholamjiak, W. *Fixed Point Optimization Algorithms and Their Applications*. Morgan Kaufmann, Cambridge, MA, 2024.

[9] Combettes, P. L.; Glaudin, L. Quasinonexpansive iterations on the affine hull of orbits: From Mann's mean value algorithm to inertial methods. *Appl. Anal.* **96** (2017), 1451–1472.
[10] Cortes, C.; Vapnik, V. Support-vector networks. *Machine Learning* **20** (1995), no. 3, 273–297.
[11] Cover, T.; Hart, P. Nearest neighbor pattern classification. *IEEE Trans. Inform. Theory* **13** (1967), no. 1, 21–27.
[12] Cubuk, E. D.; Zoph, B.; Mane, D.; Vasudevan, V.; Le, Q. V. AutoAugment: Learning augmentation strategies from data. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, 2019, 113–123.
[13] Cox, D. R. The regression analysis of binary sequences. *J. Roy. Statist. Soc. Ser. B* **20** (1958), no. 2, 215–232.
[14] Dong, Q. L.; Lu, Y. Y.; Yang, J. The extragradient algorithm with inertial effects for solving the variational inequality. *Optim.* **65** (2016), 2217–2226.
[15] Dong, Q. L.; Shehu, Y.; Li, X. H. A new inertial Tseng's extragradient method for monotone inclusion problems. *J. Comput. Appl. Math.* **348** (2019), 105–117.
[16] Gatys, L. A.; Ecker, A. S.; Bethge, M. Image style transfer using convolutional neural networks. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, 2016, 2414–2423.
[17] Geurts, P.; Ernst, D.; Wehenkel, L. Extremely randomized trees. *Machine Learning* **63** (2006), no. 1, 3–42.
[18] Goodfellow, I.; Pouget-Abbady, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Red Hook, NY, 2014, 2672–2680.
[19] Han, H.; Wang, W. Y.; Mao, B. H. Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning. In *Proc. Int. Conf. Intelligent Computing*, Berlin, Springer, 2005, 878–887.
[20] Huang, G. B.; Zhu, Q. Y.; Siew, C. K. Extreme learning machine: Theory and applications. *Neurocomputing* **70** (2006), 489–501.
[21] Iyiola, O. S.; Shehu, Y. Convergence results of two-step inertial proximal point algorithm. *Appl. Numer. Math.* **182** (2022), 57–75.
[22] John, G. H.; Langley, P. Estimating continuous distributions in Bayesian classifiers. In *Proc. 11th Conf. Uncertainty in Artificial Intelligence*, 1995, 338–345.
[23] Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.-Y. LightGBM: A highly efficient gradient boosting decision tree. In *Proc. 31st Int. Conf. Neural Information Processing Systems*, 2017, 3149–3157.
[24] Kim, T. H.; Xu, H. K. Strong convergence of modified Mann iterations with asymptotically nonexpansive mappings and semigroups. *Nonlinear Anal.* **64** (2006), 1140–1152.
[25] Krizhevsky, A.; Sutskever, I.; Hinton, G. E. ImageNet classification with deep convolutional neural networks. In *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Red Hook, NY, 2012, 1097–1105.
[26] LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **86** (1998), 2278–2324.
[27] Li, S.; Tan, B.; Zhou, Z. A modified inertial hybrid and shrinking projection algorithm for solving fixed point problems. *Mathematics* **8** (2022), 236.
[28] Maingé, P. E. Convergence theorems for inertial KM-type algorithms. *J. Comput. Appl. Math.* **219** (2008), 223–236.
[29] Nakajo, K.; Takahashi, W. Strong convergence theorems for nonexpansive mappings and nonexpansive semigroups. *J. Math. Anal. Appl.* **279** (2003), 372–379.
[30] Polyak, B. T. *Introduction to Optimization*. Optimization Software Inc., New York, 1987.
[31] Quinlan, J. R. Induction of decision trees. *Machine Learning* **1** (1986), no. 1, 81–106.
[32] Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, 2016, 779–788.
[33] Rockafellar, R. T. Monotone operators and the proximal point algorithm. *SIAM J. Control Optim.* **14** (1976), 877–898.
[34] Schizas, C.; Theumann, N.; Burn, A.; Tansey, R.; Wardlaw, D.; Smith, F. W.; Kulik, G. Qualitative grading of severity of lumbar spinal stenosis based on the morphology of the dural sac on magnetic resonance images. *Spine* **35** (2010), 1919–1924.
[35] Shukla, S.; Pant, D. R.; Sinkala, Y. A general Picard–Mann iterative method for fixed point problems and its application. *Symmetry* **14** (2022), 1741.
[36] Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint* **arXiv:1409.1556** (2015).
[37] Tan, B.; Zhou, Z.; Li, S. Strong convergence of modified inertial Mann algorithms for nonexpansive mappings. *Mathematics* **8** (2020), 462.
[38] Takahashi, W. *Nonlinear Functional Analysis*. Yokohama Publishers, Yokohama, 2000.
[39] Takahashi, W.; Takeuchi, Y.; Kubota, E. Strong convergence theorems by a shrinking projection method. *J. Math. Anal. Appl.* **341** (2008), 276–286.
[40] Wang, Z.; Bovik, A. C.; Sheikh, H. R.; Simoncelli, E. P. Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing, **13(4)** (2004), 600-612.
[41] Yao, Y.; Liou, Y. C.; Kang, C. M. Strong convergence of shrinking projection methods for a family of quasi-nonexpansive mappings. *Appl. Math. Comput.* **218** (2012), 6938–6946.

[42] Zhou, Z.; Tan, B.; Li, S. An inertial shrinking projection algorithm for split common fixed point problems. *J. Appl. Anal. Comput.* **9** (2019), 1040–1056.

[1] School of Science, University of Phayao, Phayao 56000, Thailand
Email address: papinwich.ps@hotmail.com
Email address: prasit.ch@up.ac.th
Email address: watcharaporn.ch@up.ac.th

[2] Advanced Numerical Optimization Research Group, Department of Mathematics
Email address: mint11871@hotmail.com

[3] School of Information and Communication Technology, University of Phayao 56000, Thailand
Email address: 65024300@up.ac.th
Email address: 65024333@up.ac.th
Email address: kanokwatt.sh@up.ac.th

[4] Center for General Education, China Medical University, Taichung 40402, Taiwan
Email address: yaojc@mail.cmu.edu.tw

[5] Research Center in Optimization and Computational Intelligence for Big Data Prediction, Department of Mathematics, Faculty of Science, Chiang Mai University, Chiang Mai 50200, Thailand
Email address: suthep.s@cmu.ac.th