INTERNATIONAL JOURNAL ON SMART SENSING AND INTELLIGENT SYSTEMS

Research Article | DOI: 10.2478/ijssis-2026-0030 Issue 1 | Vol. 19 (2026)

Enhanced Skill Optimization Algorithm and Stacked Long Short-Term Memory with Sech Activation Function for Gastrointestinal Disease

Janagama Srividya[1,*], and Harikrishna Bommala[2]

[1]Department of Computer Science & Engineering, Bharatiya Engineering Science & Technology Innovation University, Gorantla, Andhra Pradesh, India

[2]Department of Computer Science & Engineering, KG Reddy College of Engineering & Technology, Hyderabad, India

*E-mail: janagamasrividya@gmail.com

Received for publication August 07, 2025.

Abstract

Gastrointestinal (GI) disease significantly affects the quality of human life, leading to an increased death rate. Deep learning (DL) is an emerging tool that helps analyze medical images for disease diagnosis, prediction, and treatment planning. However, existing DL methods fail to accurately detect rare GI diseases due to insufficient feature representation, irrelevant feature selection, and high positive rates, which limit diagnostic reliability. This research proposes the Enhanced Skill Optimization Algorithm and Stacked Long Short-Term Memory with Sech Activation Function to efficiently select the relevant features and capture complex patterns in GI disease diagnosis. In traditional SOA, the Gorilla Troop Optimizer is used as an enhanced strategy in the exploration phase to select relevant features by diversifying the search space, which prevents premature convergence, minimizes dimensionality, and enhances generalization. The Stacked LSTM enhances the model's ability to capture intricate temporal dependencies, whereas the Sech activation function provides smooth gradient flow that minimizes vanishing and exploding gradients during training. Advanced preprocessing and deep feature extraction methods are used to enhance the model's ability to accurately capture significant patterns. Hence, the proposed method achieves superior accuracies of 99.60% and 99.88% on the Kvasir V1 and V2 dataset, respectively, compared with existing methods such as SK-Net, demonstrating high potential for accurate GI diagnosis and offering a robust solution for clinical image analysis.

Keywords

deep learning, enhanced skill optimization algorithm, gastrointestinal disease, Gorilla troop optimizer, stacked long short-term memory with Sech activation function

I. Introduction

Gastrointestinal (GI) infections lead to serious problems in diagnosis and are a major cause of the high death rate in humans due to various infectious diseases related to the digestive system [1]. Thus, endoscopic examinations are the first process in the common diagnosis of GI tract issues, effectively identifying disorders [2]. Endoscopic analyses are used to identify GI diseases at an early stage. These analyses improve the identification and classification of diseases by understanding their medical features [3]. Consequently, the quality of GI images that achieve accurate detection of lesions depends on the operator's skill level and the effort of endoscopists [4]. In the case of GI diseases, several deep learning (DL) and machine learning (ML)

Open Access. Professor Subhas Chandra Mukhopadhyay. CC BY-NC-ND
© 2026 Srividya and Bommala, This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivs 4.0 License https://creativecommons.org/licenses/by-nc-nd/4.0/

approaches have been developed to detect and classify upper and lower intestinal, gastric, and combined diseases [5]. In particular, large-scale datasets provide substantial data for model training; however, challenges remain due to variable noise and image quality. The preprocessing stage improves image quality and reduces unwanted noise, followed by feature extraction to capture high-level diagnostic information [6, 7]. Furthermore, feature selection is necessary to identify relevant features by addressing barriers such as the instability of peptides in gastric acid and intestinal enzymes [8, 9].

Despite these advances, several challenges remain in applying DL for reliable medical diagnosis [10]. Malignant GI disease diagnosis remains a common target, but ML has struggled with complex patterns in large datasets [11]. Although DL effectively manages large datasets depending on image size and quality [12], for GI disorders such as abnormal tissues and polyps, DL-based methods have enhanced diagnostic efficiency by enabling better classification of non-cancerous and cancerous images [13–15]. However, existing DL methods fail to accurately detect rare GI diseases due to insufficient feature representation, selection of irrelevant features, and high positive rates, which limit diagnostic reliability and clinical applicability. To solve this issue, the Enhanced Skill Optimization Algorithm (ESOA) and Stacked Long Short-Term Memory with Sech Activation Function (Stacked LSTM-SAF) are proposed as novel methods for diagnosing GI disease. Unlike the traditional SOA, this integration employs ESOA's adaptive skill-based exploration with the gorilla troop optimizer's (GTO) efficient global search ability to select highly relevant features while minimizing noise and enhancing feature representation. The Stacked LSTM improves upon traditional LSTM and hybrid DL models such as Convolutional Neural Network (CNN)-LSTM by capturing more complex spatial and temporal patterns in endoscopic images due to its multilayer structure. Unlike single-layer LSTM and CNN-LSTM, which focus on either spatial or temporal features, the Stacked LSTM effectively integrates both, enhancing model accuracy in detecting rare GI diseases.

The main contributions of this paper are as follows:

- Adaptive histogram equalization (AHE) enhances the contrast of GI images, whereas the bilateral filter minimizes noise, improving image quality for accurate performance.
- DenseNet201 captures deep hierarchical features by enabling feature reuse and minimizing the vanishing gradient issue, while EfficientNet-B3 balances model depth, resolution, and width to extract rich and representative features that enhance overall classification accuracy.
- In the SOA, the GTO is included in the exploration stage to diversify the search space, enabling the selection of highly relevant features, minimizing dimensionality, avoiding premature convergence, and enhancing generalization for accurate GI disease diagnosis.
- The Stacked LSTM with the Sech activation function effectively captures intricate temporal dependencies, while the Sech function ensures smooth gradient flow, minimizing exploding gradients during training. This combination improves learning effectiveness and enhances the model's ability to accurately classify rare GI diseases.

In summary, the limitations of existing DL methods in detecting rate GI diseases represent a critical research gap that must be addressed to improve diagnostic accuracy and clinical reliability. The proposed ESOA with Stacked LSTM-SAF addresses this gap by enhancing feature selection, capturing intricate spatial-temporal patterns, and stabilizing training in endoscopic images. By developing methodological innovations with practical applicability, this research advances accurate GI disease diagnosis and supports clinicians in achieving rapid and accurate performance.

The remaining sections are organized as follows: Section II presents the literature survey, Section III explains the proposed methodology, Section IV analyzes the experimental results, and Section V draws the conclusion.

II. Literature Survey

This research reviews GI image classification methods by organizing the literature survey into three subsections: CNN-based methods, GAN-based methods, and Capsule network (CN)-based methods. Each category presents key techniques with their advantages and limitations, offering a clearer understanding.

a. CNN-based methods

Farah Mohammad and Muna Al-Razgan [16] presented a cosine support vector machine (CSVM) along with CNN-based Inception V3 and DenseNet201 for feature extraction to identify stomach disease. This approach combined feature effusion and

concatenation using parallel maximum covariance to perform feature fusion. The CSVM effectively handled the high-dimensional feature vectors extracted by CNN, ensuring accurate classification. However, CSVM struggled with poor scalability on large datasets due to computing pairwise cosine similarity, which affected model efficiency and reduced classification performance.

Sharma et al [17] developed an LPNet–CNN with discrete wavelet transformation (DWT)-based pooling in place of conventional max and average pooling to automatically classify polyps and non-polyps. The DWT enabled multi-resolution analysis, allowing LPNet-CNN to capture both global and local features effectively, thereby improving classification accuracy. However, LPNet was sensitive to noise and texture variations because wavelet decomposition amplified high-frequency components, leading to overlap with noise and misclassification.

Naz et al. [18] implemented a hybrid feature model that fine-tuned the ResNet18 model and evaluated it using XcepNet23 and local binary pattern (LBP) features combined with the CNN method. The LBP excelled at capturing features, making the hybrid approach more robust to variations in medical images. ResNet18 and XceptNet23 captured high-level features, while LBP focused on low-level spatial patterns. Nevertheless, ResNet18, having relatively shallow depth, struggled to capture highly complex features, leading to inaccurate performance.

Intissar Dhrari Hajsalem et al. [19] presented an encoder-decoder network (EDN) to detect GI disease. Initially, the Region of Interest was detected by utilizing the encoder and decoder network to select appropriate and significant information. Pre-trained methods such as ResNet50, VGG16, InceptionV3, VGG19, and support vector machine (SVM) were employed to extract features and accurately classify GI diseases. The EDN consumed more time but achieved better results by eliminating inappropriate information and effectively determining pertinent features. However, EDN lost fine-grained spatial information because the encoding process involved successive downsampling and pooling operations that minimized feature map resolution.

Samra Siddiqui et al. [20] established SNet to classify GI disorders using endoscopic images. Initially, endoscopic images were preprocessed using resizing and augmentation to ensure uniform image dimensions and enhance image quality. The Minimum Redundancy Maximum Relevance was used to select features, thereby enhancing model performance. However, SNet had limited ability to capture complex and subtle features because its shallow feature extraction layers could not sufficiently model complex variations and textures.

b. GAN-based method

Hyun-Cheol Park et al. [21] suggested a Star-Generative Adversarial Network (Star-GAN) aimed at resolving challenges in learning mappings across multiple domains by integrating the multi-translation with InceptionNet-V3 method. The Star-GAN handled multiple domains with a signal generator and discriminator, significantly reducing complexity compared with traditional GAN. The InceptionNet-V3 extracted features and captured intricate patterns in the data, ensuring improved classification performance. The Star-GAN was efficient in capturing complex dependencies but lacked model diversity, as InceptionNet-V3 performance depended on the quality of extracted features. However, Star-GAN generated unrealistic images due to heavy reliance on limited data, which restrained the model's ability to learn diverse and accurate features.

c. CN

Henrietta Adjei Pokuaa et al. [22] introduced a CN to recognize GI conditions. Although the CN achieved low performance on complex dataset, it was enhanced by modeling complex dependencies between features, which benefited the classification of GI images and enabled the extraction of richer information from the GI disease image. Nevertheless, CN was sensitive to small variations because it encoded spatial relationships and posed information about features; hence, minor distortions misaligned vectors, leading to incorrect feature routing among capsules and reduced accuracy.

From the overall analysis, existing techniques had limitations such as poor scalability, sensitivity to noise and texture variations, difficulty in capturing highly complex features, and failure to select relevant features, leading to insufficient representation and reduced accuracy. Hence, this research proposes ESOA with Stacked LSTM-SAF to classify GI diseases effectively. ESOA selects the most appropriate features, minimizing the impact of irrelevant data and ensuring more informative feature representation. Moreover, the Stacked LSTM captures highly intricate temporal and spatial patterns, enhancing the model's

ability to identify subtle regions. This integration improves robustness against noise and texture variations while maintaining scalability. As a result, the proposed method achieves more accurate and reliable classification compared with existing methods.

III. Proposed Methodology

This research proposes the ESOA with Stacked LSTM using the SAF to optimize classification performance. Initially, data collected from the Kvasir V1 and V2 dataset are preprocessed using AHE to enhance image contrast and reduce noise while preserving edges by employing the Bilateral filter. DenseNet-201 is used to extract features, capturing hierarchical feature through dense connections that reduce the risk of vanishing gradients, while EfficientNet-B3 provides a balanced trade-off between performance and efficiency by scaling depth-wise features. The ESOA efficiently selects the relevant features and reduces high dimensionality, whereas the Stacked LSTM with SAF classifies the GI diseases. As shown in Figure 1, the proposed method contains five primary stages, where each stage is designed to enhance image quality, minimize dimensionality, and improve the accuracy of GI disease classification.

a. Dataset

In this section, two datasets—Kvasir V1 and V2—are considered for GI tract classification. The dataset details, including image counts and categories of GI diseases, are described below.

a.i Kvasir-V1 dataset

The Kvasir-V1 dataset [23] for GI tract analysis consists of 4,000 images categorized into eight classes, with each class containing 500 images. These images are segmented into anatomical landmarks and endoscopic polyp removal categories. For colorectal classification, the dataset focuses on two primary classes: polyps and non-polyps, which are utilized for training and testing the classification model.

a.ii Kvasir-V2 dataset

The Kvasir-V2 dataset [24] contains eight pathological findings, each represented by 1,000 images. For the experiment, only five classes were utilized, as the remaining three classes were repetitions of the selected ones. The dataset images were redistributed using an 80:20 split for training and testing and resized to a $32 \times 32 \times 3$ image size.

b. Pre-processing

After collecting the images, the preprocessing stage is performed using AHE and the Bilateral filter [25] to enhance image quality and minimize noise. A detailed description of these methods is provided below.

AHE: addresses this issue by using a unique histogram for each pixel in the image. This histogram is calculated based on the intensity values

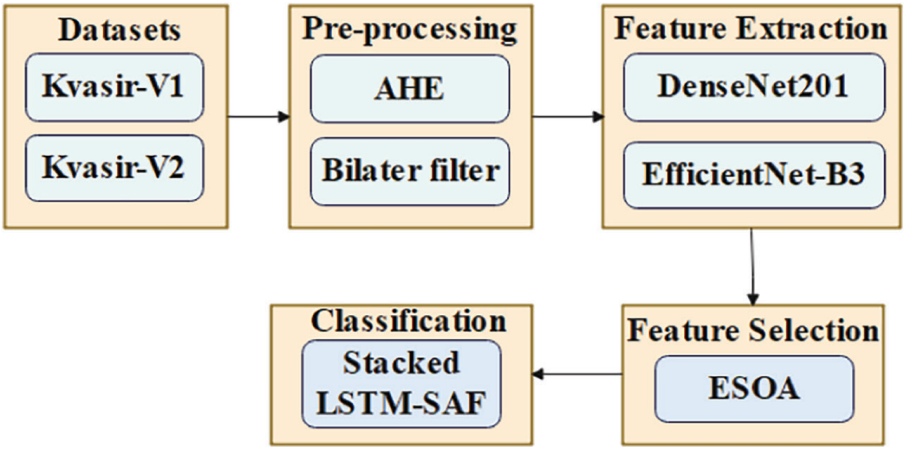


Figure 1: Workflow of proposed methodology. AHE, adaptive histogram equalization; ESOA, enhanced skill optimization algorithm; Stacked LSTM-SAF, stacked long short-term memory with Sech activation function.

within a surrounding window, known as the contextual region. Compared with HE, AHE provides better performance as it effectively addresses contrast enhancement. It effectively controls the level of contrast enhancement in the output image, especially when the original image has low contrast, where most values are concentrated in the middle of the grayscale image.

Bilateral filter: This method filters the image based on both range and pixel values. It is a local, non-linear, and non-iterative technique that considers gray-level similarities and neighboring pixels to enhance image contrast by preserving fine information and local structures. This approach effectively addresses issues of image noise and varying brightness by combining AHE and bilateral filtering to enhance contrast. The process involves applying AHE to each color channel of the input image, followed by bilateral filtering on the equalized channels. The primary goal of the bilateral filter is to preserve edges while filtering the image. The filtered channels are then recombined into a unified RGB image, and normalization procedures are applied to enhance overall contrast. The bilateral filter is mathematically expressed in Eq. (1), where the image intensity at a pixel position P is defined I_P, where G_σ denoted as a 2D Gaussian kernel. The normalization factor, G_P ensures that the sum of pixel weights equals 1.0. The parameters s & r control the extent of filtering applied to the image, as represented in Eqs (2)–(4) for the HE process, which modifies image intensities to improve contrast:

$$BF[I]_P = \frac{1}{W_p}\sum_{q\varepsilon s} G_{\sigma_s}\left(\|P-Q\|\right)G_{\sigma_s}\left(|I_p - I_q|\right)I_q \tag{1}$$

$$W_P = \sum_{q\varepsilon s} G_{\sigma_s}\left(\|P-Q\|\right)G_{\sigma_s}\left(|I_p - I_q|\right)I_q \tag{2}$$

$$I_i, j = \text{floor}(L-1)\sum_{n=0}^{f_i,j} P_n \tag{3}$$

$$P_n = \frac{I_n}{T_P}\ n = 0,1,\ldots..,L-1 \tag{4}$$

Here, L denotes the count of possible intensity values, I_n represents the number of pixels with a intensity, and n and T_P indicate the total number of pixels. These two strategies—AHE and the Bilateral filter—are utilized to enhance image quality and remove noise in GI images. The preprocessed images are then fed into the feature extraction to extract the high-level feature. Figure 2 represents the sample original and preprocessed images.

c. Feature extraction

The preprocessed input is fed into the feature extraction using DenseNet-201 and EfficientNet-B3 to extract high-level features from GI images. Deep CNN models are recognized for their robust classification performance in GI images. Their layers have direct connections to the subsequent layers, which enhance the learning rate while minimizing information loss. Compared with other CNN models, DenseNet-201 network requires fewer parameters. The model ensures high information retention with minimal loss from the first to the last layer and incorporates a gradient function to reduce risk of overfitting.

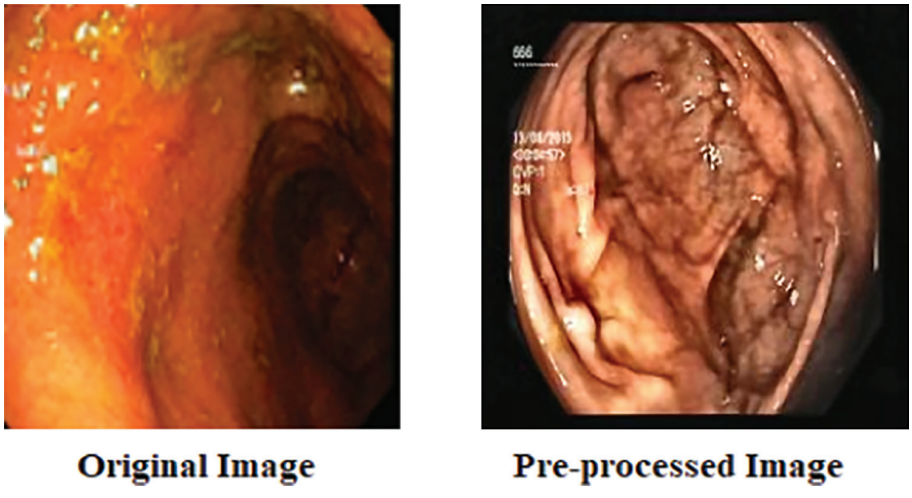


Figure 2: Sample original and preprocessed image.

DenseNet-201: The input size is set to $224 \times 224 \times 3$ with 201 structured layers organized into four dense blocks. A global average pooling layer produces a 1920-dimensional feature vector for feature extraction. The GI images ζ_z are used as input to DenseNet-201 [26], which consists of M layers and transformation filters denoted by the non-linear function $S_m(.)$. The transformation filter $s_m(.)$ is a concatenated function that includes convolution, batch normalization, pooling, and rectified linear unit (ReLU). In a classical CNN model, the output of m^{th} layer serves as the input to $(m+1)^{th}$ layer, mathematically represented in Eqs (5) and (6):

$$B_m = S_m(B_m - 1) \tag{5}$$

$$b_n = S_m([b_0, b_1, \ldots\ldots, b_{n-1}]) \tag{6}$$

Here, each layer is directly connected to every other layer, and m^{th} layer carries information evaluated from all network layers $b_0, b_1, \ldots\ldots, b_{n-1}$. The computed feature map of layer $0, \ldots\ldots n-1$ is defined in Eq. (6).

EfficientNet-B3: This method processes input images of size 300×300 using compound scaling across width, depth, and resolution to balance accuracy and efficiency. It adjusts these characteristics using predetermined factors, and the member of this group amplifies the equilibrium by providing significant generalizability. Initially, the input image size of 600×600 pixels is compared with that of EfficientNet-B2, which contains 10.7 million parameters. It includes three levels replaced with additional layers for the use case of GI image classification within the subsets of classes. Furthermore, these layers were modified by removing and incorporating additional dense, batch normalization, and dropout layers. A global average pooling layer produces a 1536-dimensional feature vector. Finally, a dense layer with SoftMax activation is used for multi-class classification, effectively capturing complex patterns. Both DenseNet-201 and EffcientNet-B3 [27] are fine-tuned using the Adam optimizer with a batch size of 32, a learning rate of 0.001, and the cross-entropy loss function. The features extracted from DenseNet-201 and EfficientNet-B3 are concatenated to form a joint representation, which is then fed into the feature selection using ESOA [28].

d. Feature selection

After extracting features, the ESOA is applied to select the most relevant features. The SOA is chosen for its ability to mimic human skill learning, allowing efficient exploration and exploitation for feature selection in high-dimensional data. ESOA is more efficient in navigating complex and high-dimensional search spaces due to the enhanced exploration and exploitation capabilities of the GTO. Compared with existing methods such as the osprey optimization algorithm (OOA), coati optimization algorithm (COA), bald eagle search optimizer (BES), and SOA, the ESOA improves convergence speed and solution accuracy by incorporating GTO. Algorithms such as OOA, COsA, BES, and SOA suffer from premature convergence, which limits their ability to identify global optima. By integrating GTO into SOA, exploration ability is enhanced through diverse search behavior inspired by gorilla leadership dynamics. This integration avoids local trapping, enhances feature relevance, and ensures better generalization. Thus, ESOA combines SOA's adaptive learning with GTO's robust exploration capability to achieve superior feature selection and higher diagnostic accuracy. The initialization process of ESOA is defined in Eqs (7) and (8):

$$X = \begin{bmatrix} X_1 \\ . \\ . \\ . \\ X_2 \\ . \\ . \\ . \\ X_N \end{bmatrix}_{N \times m} = \begin{bmatrix} X_{1,1} X_{1,d} X_{1,m} \\ . \\ . \\ . \\ X_{i,1} X_{i,d} X_{i,m} \\ . \\ . \\ . \\ X_{N,1} X_{N,d} X_{N,m} \end{bmatrix}_{N \times m} \tag{7}$$

Here, X represents the SOA candidate solution, X_i denotes the *i*th candidate solution, and $X_{i,d}$ indicates the *i*th variable corresponding to the *i*th solution of SOA. N refers to the population count, while m denotes the dimension.

$$F = \begin{bmatrix} F_1 \\ . \\ . \\ . \\ F_2 \\ . \\ . \\ . \\ F_N \end{bmatrix}_{N \times m} = \begin{bmatrix} F(X_1) \\ . \\ . \\ . \\ F(X_i) \\ . \\ . \\ . \\ F(X_N) \end{bmatrix}_{N \times 1} \tag{8}$$

The arbitrary assignment covers all possible areas in the given $N \times m$ search space, and the objective

function is evaluated for each individual member in the issue variables. The OF vector is expressed in Eq. (8), where F_i represents the ith member values of the OF, and F denotes the set of all members as an objective vector. The population is updated in two stages: exploration and exploitation. The second phase, based on skill enhancement, involves distinct activities and efforts.

d.i Mutation strategy

This approach employs a two-mutation operation used to generate a mutated vector, which includes conventional mutation and current-to-best mutation, as expressed in Eq. (9):

$$v_i^G = X_{r_1}^G + F.\left(X_{r_1}^G - X_{r_2}^G\right) + F.\left(X_{r_3}^G - X_{r_4}^G\right), r_1 \neq r_2 \neq r_3 \neq i \tag{9}$$

$$v_i^G = X_{\text{best}}^G + F.\left(X_{r_1}^G - X_{r_2}^G\right) + F.\left(X_{r_3}^G - X_{r_4}^G\right) \tag{10}$$

Here, $r_1, r_2,$ and r_3 are random indices, where $r_1, r_2, r_3 \in \{1,2,\ldots N\} cc$. The F denotes the scaling factor within the range of [0, 2], and the current-to-best mutation vector is represented in Eq. (10) The r_4 denotes the random index, v_i^G represents the mutated vector for the ith candidate solution in generation, $X_{r_1}^G, X_{r_2}^G, X_{r_3}^G, X_{r_4}^G$ refers to randomly selected candidate solution in the population at generation G, and X_{best}^G determines the best candidate solution in population at G. The fitness function primarily consists of two components: the count of features and the classification accuracy. It is designed to minimize classification error, where small error values indicate better fitness scores, as defined in Eq. (11):

$$\text{Fitness Function} = w \times \propto \left(1-w\right) \times \frac{|s|}{|d|} \tag{11}$$

Here, $|d|$ indicates the toal count of features in the dataset, $|s|$ represents the number of selected features, $\propto$ denotes the classification error for the selected feature subset, and $w \in [0,1]$ signifies the relative weight values assigned to the classification error and feature count.

d.ii Exploration strategy-based GTO

The GTO simulates the lifestyle and social behavior of gorillas in forests. GTO incorporates unique exploitation strategies. One such strategy involves group members following the dominant leader gorilla, where the silverback represents the finest solution. This is expressed in Eqs (12)–(15):

$$GX(t+1) = L \times M \times \left(X(t) - X_{\text{silverback}}\right) + X(t) \tag{12}$$

$$M = \left(\left|\sum_{i=1}^{N} X_i(t)/N\right|^{2^L}\right)^{\frac{1}{2^L}} \tag{13}$$

$$L = C \times l \tag{14}$$

$$C = \left(\cos\left(2 \times r_5\right) + 1\right) \times \left(1 - \frac{t}{T_{\max}}\right) \tag{15}$$

Here, l represents a random value in the range [–1, 1], and $T_{\max}$ denotes the total number of iterations. $GX(t+1)$ represents the updated position of a gorilla at iteration t+1, $X(t)$ denotes the current position of the gorilla, $X_{\text{silverback}}$ determines the position of the dominant leader gorilla, L illustrates the adaptive coefficient that influences step size, and M provides the influence factor for the gorilla's movement within the search space.

d.iii Exploitation

In this section, each member strives to enhance its skill through continuous activity and practice. The SOA notion formulates a local search (lS) to improve exploitation, where every member explores solutions within its neighborhood to enhance objective values. The most suitable candidate solution expands the OF, and this secondary phase is expressed in Eqs (16) and (17):

$$X_i^{P^2} : X_{i,d}^{P2} = f(x) = \begin{cases} x_{i,d} + \dfrac{1-2r}{t} \times x_{i,d}, x < 0.5 \\ x_{i,d} + \dfrac{lb_j - r\left(ub_j - lb_j\right)}{t}, \text{else} \end{cases} \tag{16}$$

$$X_i = \begin{cases} X_i^{P2}, F_i^{P2} < F_i \\ X_i, \text{else} \end{cases} \tag{17}$$

Here, $X_i^{P^2}$ denotes the updated position of the ith candidate after the second step, $x_{i,d}^{p2}$ represents the current position of the ith candidate in the dth dimension, F_i^{p2} indicates the recent objective value, and t denotes the number of iterations. lb_j and ub_j define the boundaries of the jth variable, r illustrates a random number, t represents the current iteration number, lb_j denotes the lower bound of the j_{th} variable, and ub_j denotes the upper bound of the j_{th}

variable. Figure 3 demonstrates the workflow of the ESOA method.

e. Classification

In this section, GI disease classification is performed using the Stacked LSTM-SAF [29], which learns and predicts patterns in sequential data and has the ability to process entire data sequences efficiently. The Stacked LSTM provides a hierarchical structure that captures long-term dependencies more effectively compared with existing methods such as the recurrent neural network (RNN), LSTM, and gated recurrent unit (GRU). Unlike GRU and single-layer LSTM models, the Stacked LSTM learns deeper temporal patterns and intricate feature representations by integrating multiple layers. Existing models such as RNN, GRU, and LSTM are limited in handling complex sequential dependencies and lack robustness and scalability to capture fine-grained variations. Hence, the Stacked LSTM offers improved accuracy and generalization for sequence modeling tasks. LSTM uses cells and gates to control the flow of information, preventing the loss of short-term memory. An LSTM layer consists of memory blocks with input, output, and forget gates, as well as one or more memory cells coupled in loops within each block. These three multiplicative units mimic the actions of resetting memory cells. The input, output, and forget gates are evaluated using Eqs (18)–(20):

$$G_i = \sigma\left(y_t W_{yi} + S_{t-1} W_{si} + \beta_i\right) \tag{18}$$

$$G_f = \sigma\left(y_t W_{yf} + S_{t-1} W_{sf} + \beta_f\right) \tag{19}$$

$$G_o = \sigma\left(y_t W_{yo} + S_{t-1} W_{so} + \beta_o\right) \tag{20}$$

The input node of the memory cell is evaluated using Eqs (21)–(23), while the internal and hidden

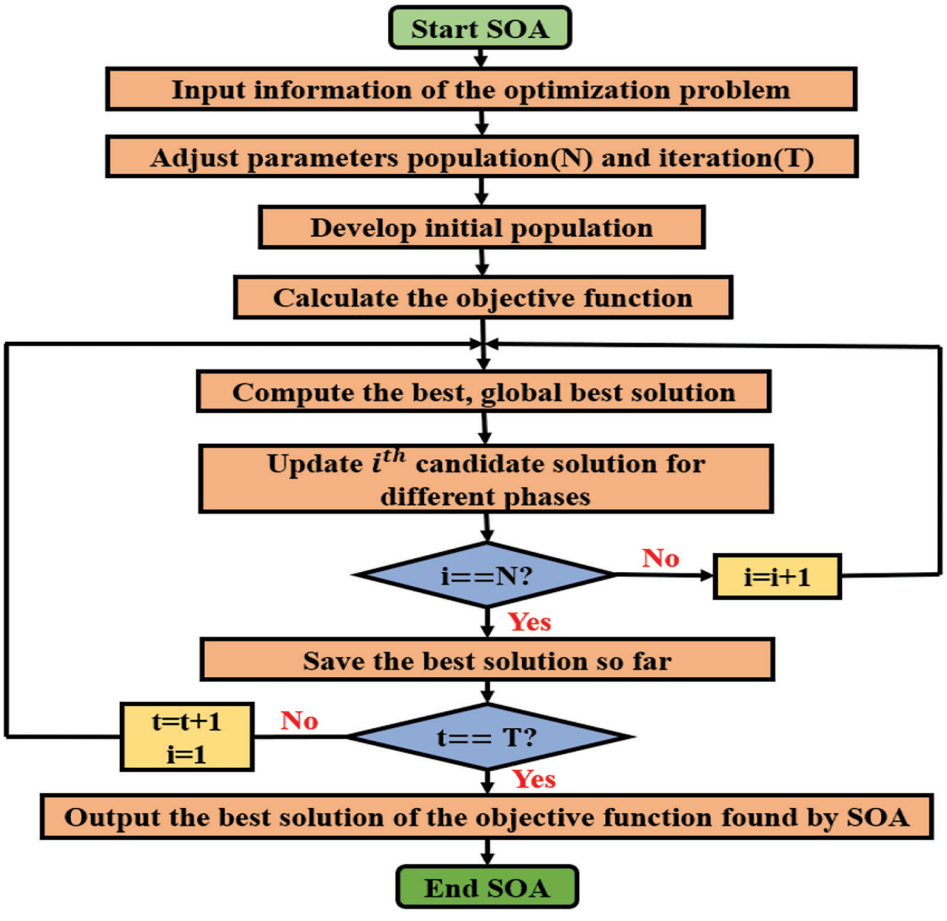


Figure 3: Workflow of the ESOA method. ESOA, enhanced skill optimization algorithm.

states of memory cell are evaluated by the activation function:

$$\tilde{\theta}_t = \tan h\,(y_t W_{yc} + S_{t-1} W_{sc} + \beta_c) \tag{21}$$

$$\theta_t = G_f \odot \theta_{t-1} + G_i \odot \tilde{\theta}_t \tag{22}$$

$$S_t = G_o \odot \tan h(\theta_t) \tag{23}$$

Here, t denotes the time step, and y_t indicates the input to the Stacked LSTM model. In the Stacked LSTM model, multiple hidden layers are employed instead of a single layer, allowing parameters to be distributed across the model to accelerate processing and better reprsent raw data using the Sech activation function. The Sech activation function is chosen because it provides smoother gradient flow, which helps prevent vanishing and exploding gradients compared with the ReLU and Sigmoid. Unlike ReLU, which suffers from the "dying neuron" problem caused by zero gradients for negative inputs, Sech remains symmetric and continuously differentiable around zero, ensuring stable updates. In contrast to the Sigmoid, which squashes values between 0 and 1—leading to saturation—Sech manages a wider dynamic range, thereby minimizing the risk of gradient information. Overall, Sech improves model stability and convergence speed. The mathematical formulation of the Sech activation function is expressed in Eqs (24) and (25):

$$f(a)_i = a_i \operatorname{sech}(a_i) \tag{24}$$

$$a = b + W^t x \tag{25}$$

Here, a_i represents the function expressed in Eq. (24), w indicates the input weight, x represents the input, and b denotes the bias. This activation function is shown in a ordination system, where x lies in the range (–10, 10). The Sech is symmetric around zero and bounded between 0 and 1, similar to the sigmoid, which helps maintain stability during gradient-based optimization. It is defined for all integers and is fully differentiable and odd-symmetric with respect to the origin, which is essential for optimization and achieving better convergence. The hyperparameters of the LSTM-SAF are optimized using grid search to achieve optimal training performane. This setup employs a learning rate of 0.001, the Adam optimizer, Sech activation function, a batch size of 64, a weight decay of $1e^{-4}$, and a maximum of 50 epochs with early stopping (patience = 8) to effectively learn patterns from the data. Algorithm 1 presents the pseudocode of the proposed method to ensure reproducibility. Figure 4 depicts the structure of the Stacked LSTM, and Figure 5 shows sample images of actual and predicted labels for (A) Kvasir-V1 (B) Kvasir-V2.

Algorithm 1:

Input: Dataset {Kvasir-V1 and Kvasir-V2}
Output: Predicted labels
Step 1: Preprocessing
  For each image in the dataset
    Apply AHE for contrast enhancement
    Employ the Bilateral filter to minimize noise
  End for
Step 2: Feature extraction
  Extract features using DenseNet-201
  Extract features using EfficientNet-B3
  Concatenate and combine the feature vectors
Step 3: Feature selection
  Initialize the population of candidate feature subsets
  For iteration = 1 to MaxIteration do
    For each candidate solution do
      Evaluate the fitness function
  End for
  Generate mutated vectors using the mutation strategy
  Update candidate solutions based on leader guidance using GTO during the exploration stage
  Perform local search to refine selected features during the exploitation stage
  End for
  Select the best feature subset
Step 5: Classification
Define the Stacked LSTM model
  Include multiple LSTM layers
  Employ the Sech activation function
  Apply Softmax for multiclass classification
Predicted label

IV. Experimental Results

In this research, the ESOA with Stacked LSTM using SAF technique is simulated using MATLAB (MathWorks, USA) R2020b. The experiments are conducted on a system equipped with an Intel i5 processor, 64 GB Random Access Memory, 6 GB Graphics Processing Unit, and 1 TB SSD, running on the Windows 10 operating system. These specifications are sufficient to efficiently handle training and inference for Kvasir-V1 and Kvasir-V2 dataset. The model achieves convergence within a reasonable

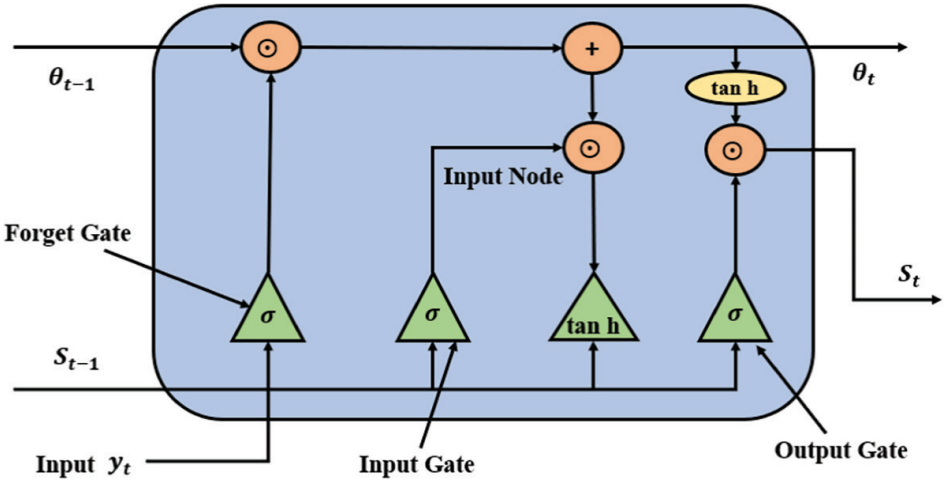
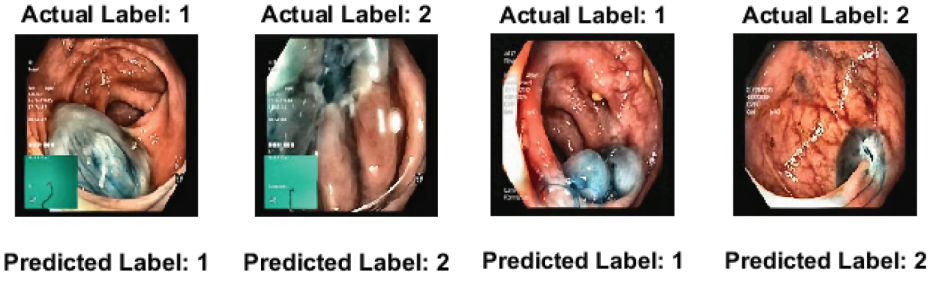


Figure 4: Structure of the stacked LSTM. Stacked LSTM, stacked long short-term memory.



(A) KVASIR-V1 (B) Kvasir-V2

Figure 5: Sample images of actual and predicted labels **(A)** Kvasir-V1 **(B)** Kvasir-V2 dataset.

training time, indicating that it is effectively trained. For GI disease classification, accuracy represents the overall measure of correct predictions in the dataset. Precision denotes the reliability of the model in identifying disease cases without misclassifying healthy instances, which is crucial to avoid unnecessary treatments. Recall determines the model's ability to correctly identify actual disease cases, thereby minimizing missed diagnoses. The F1-score, a combination of precision and recall, balances both aspects to provide a comprehensive measure in imbalanced datasets. These performance metrics are chosen to ensure that the model is both reliable and accurate, effectively classifying diseases while minimizing False Positives and False Negatives. The mathematical formulations of these metrics are provided in Eqs (26)–(29). The dataset is divided into 80% training and 20% testing using a random split to preserve class balance. To ensure reliability and minimize the effect of randomness from a single split, each experiment is repeated five times with different random seeds. The performance metrics are averaged across these trials to provide a more consistent and robust evaluation of the proposed method:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{26}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{27}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{28}$$

$$F1-\text{Score} = \frac{2TP}{2TP + FP + FN} \tag{29}$$

Here, TP, TN represent True Positive and True Negative, respectively.

a. Performance analysis

In this section, the proposed method, involving both feature selection and classification processes, is evaluated using several performance metrics, including Accuracy, Precision, F1-score, and Recall. Table 1 presents the performance analysis of different feature selection methods. When compared with existing methods such as the OOA, COA, BES, and SOA, the proposed ESOA achieves superior accuracies of 99.60%, 99.88%, and 99.74% on the Kvasir-V1, V2, and HyperKvasir dataset, respectively. This is due to the integration of the GTO into SOA, which enhances the exploration and exploitation to escape local optima. The adaptive search mechanism enables more accurate feature selection, which directly improves classification performance. The feature selection process is independently evaluated by measuring both the objective function and fitness value, ensuring that the selection effectiveness is validated prior to being used in the classification stage. The parameter settings for the ESOA include a population size of 30, a maximum of 100 iterations, and an inertia weight of 1.00.

Table 2 indicates the performance analysis of different classification methods across three datasets. When compared with existing methods such as the RNN, GRU, and LSTM, Stacked LSTM achieves superior accuracies of 99.60%, 99.88%, and 99.74% on the Kvasir-V1, V2, and HyperKvasir dataset, respectively. This improvement is attributed to its multiple-layer structure, which captures intricate temporal dependencies and hierarchical feature representations. The Stacked LSTM enables the model to learn both long-term and short-term patterns more effectively than LSTM, thereby enhancing generalization over varying input sequences and enhance robustness.

Table 3 provides the performance evaluation of different activation functions. Compared with existing methods, the proposed method yields better performance because Sech provides smoother gradients and prevents saturation issues. This helps maintain stable training and minimizes vanishing gradient problems across multiple layers. Consequently, the model captures deep temporal dependencies with minimal information loss, enhancing overall classification accuracy. The t-test is employed to analyze whether the

Table 1: Performance analysis of different feature selection methods

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Specificity (%)
Kvasir-V1					
OOA	94.17	94.54	93.92	94.23	92.67
COA	96.66	95.15	95.38	95.27	94.38
BES	97.37	95.12	96.30	95.71	94.79
SOA	97.99	96.53	97.98	97.25	96.95
ESOA	99.60	99.20	98.71	98.96	99.88
Kvasir-V2					
OOA	94.26	94.65	93.33	94.65	93.33
COA	95.58	95.21	94.67	95.21	94.67
BES	97.51	95.18	96.84	95.18	96.84
SOA	97.51	97.18	96.84	97.18	96.84
ESOA	99.88	99.61	97.12	99.61	97.12
HyperKvasir					
OOA	94.32	94.78	93.85	94.11	92.91
COA	96.28	95.42	95.16	95.29	94.63
BES	97.14	95.88	96.41	96.14	95.37
SOA	97.92	96.71	97.65	97.18	96.82
ESOA	99.74	99.33	98.95	99.14	99.61

BES, bald eagle search optimizer; COA, coati optimization algorithm; ESOA, enhanced skill optimization algorithm; OOA, osprey optimization algorithm; SOA.

Table 2: Performance analysis of different classification methods

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Kvasir-V1				
RNN	89.44	90.51	88.90	91.42
GRU	93.80	94.03	93.59	92.14
LSTM	96.04	95.99	94.88	94.66
Stacked-LSTM	99.60	98.71	99.88	99.20
Kvasir-V2				
RNN	90.56	90.50	91.27	92.95
GRU	94.07	93.10	92.00	94.56
LSTM	97.49	96.40	95.55	95.61
Stacked-LSTM	99.88	97.93	97.12	99.61
HyperKvasir				
RNN	89.88	90.65	89.44	91.12
GRU	93.92	94.21	93.78	92.83
LSTM	96.38	96.10	95.21	95.68
Stacked-LSTM	99.74	98.95	99.33	99.14

GRU, gated recurrent unit; LSTM, long short-term memory; RNN, recurrent neural network; Stacked LSTM, stacked long short-term memory.

Table 3: Performance analysis of different activation functions

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	T-test from p-values	CI (%)
Kvasir-V1						
Stacked LSTM- ReLU	94.97	93.47	92.60	93.63	0.032	86.18
Stacked LSTM-Tanh	95.46	93.28	94.30	95.04	0.030	87.63
Stacked LSTM-Sigmoid	97.03	96.96	95.43	96.01	0.029	89.06
Stacked LSTM-sech	99.50	98.71	99.88	99.20	0.026	94.12
Kvasir-V2						
Stacked LSTM- ReLU	93.55	92.53	92.67	91.76	0.034	89.60
Stacked LSTM-Tanh	96.62	95.71	96.00	97.26	0.030	90.17
Stacked LSTM-Sigmoid	97.45	95.50	94.18	96.65	0.028	91.78
Stacked LSTM-sech	99.88	97.93	97.12	99.61	0.024	94.36
HyperKvasir						
Stacked LSTM- ReLU	94.76	93.62	92.89	93.45	0.036	87.15
Stacked LSTM-Tanh	95.84	94.71	95.25	95.07	0.034	89.36
Stacked LSTM-Sigmoid	97.28	96.64	95.81	96.22	0.031	91.58
Stacked LSTM-Sech	99.74	98.95	99.33	99.14	0.029	94.85

CI, confidence interval; ReLU, rectified linear unit; Stacked LSTM, stacked long short-term memory.

difference between the means of two groups is statistically significant, taking into account sample size and variance. The use of p-values obtained from the t-test quantifies the likelihood that observed performance differences occurred by chance, ensuring a reliable interpretation of results. Additionally, the confidence interval (CI) provides a range of values within which the true performance metric is likely to lie, serving as a measure of estimation reliability. The CI helps assess the consistency and robustness of the proposed method.

Figure 6 presents the confusion matrices of the Stacked LSTM-SAF for (A) Kvasir-V1, (B) Kvasir-V2, and C) HyperKvasir datasets. Each matrix shows the number of correctly and incorrectly classified instances across eight GI disease classes. The proposed model demonstrates high classification performance, with minimal misclassifications observed in a few instances such as normal-z-line and ulcerative-colitis. These results indicate the model's strong ability to differentiate subtle variations in GI images across all datasets.

Figure 7 depicts the receiver operating characteristics (ROC) curve analysis of the Stacked LSTM-SAF (A) Kvasir-V1, (B) Kvasir-V2, and (C) HyperKvasir datasets. All classes achieve high Area Under the Curve, demonstrating the model's robustness and strong generalization performance across dataset. Overall, these results confirm the high effectiveness of the proposed model in accurately classifying GI diseases.

Figure 8 represents the performance analysis in terms of the standard deviation of the proposed method for (A) Kvasir-V1, (B) Kvasir-V2, and (C) HyperKvasir datasets. The proposed method attains high accuracy with low variance, indicating consistent performance across all dataset. The small error bars reflect low standard deviation, which demonstrates model stability and robust classification performance.

Table 4 illustrates the computational complexity of different methods across the three datasets. Compared with existing methods, the Stacked LSTM-SAF achieves shorter training times of 2.54 s, 2.73 s, and 5.80 s. The Sech function contributes to this improvement, as its symmetric and smooth nature prevents gradient explosion and stabilizes backpropagation, resulting in rapid convergence. Moreover, the Stacked LSTM minimizes redundant iterations during training, further achieves rapid training time.

Table 5 demonstrates the cross-dataset validation results, where the model is trained on Kvasir-V1 and tested on Kvasir-V2 evaluate its generalization capability. Compared with existing methods, the Stacked LSTM-SAF achieves higher performance across all evaluation metrics, confirming its robustness against dataset variations. This result highlights the model's strong ability to generalize effectively to unseen data, ensuring reliability for practical medical applications.

b. Comparative analysis

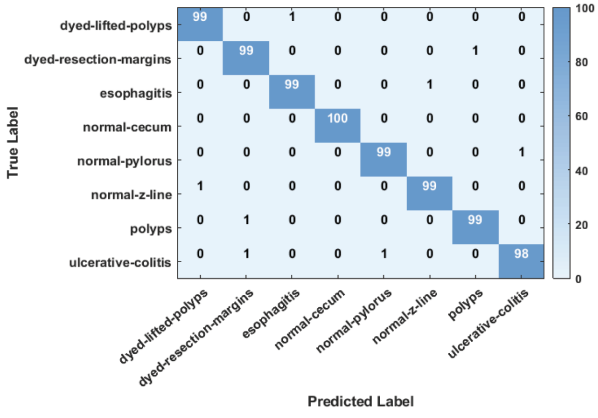
Table 6 describes the comparative analysis of the proposed method on the Kvasir-V1 and V2 datasets. When compared with existing methods in [17, 19, 20–22], the proposed ESOA with Stacked LSTM-SAF achieves higher accuracies of 99.60% and 99.88% on the Kvasir-V1 and V2 dataset, respectively. This improvement is attributed to ESOA ensuring optimal feature selection, which minimizes inappropriate data and enhances the classification. The Stacked LSTM-SAF effectively learns both shallow and deep temporal dependencies, with stable learning via SAF improving convergence, robustness, and overall classification performance.

c. Discussion

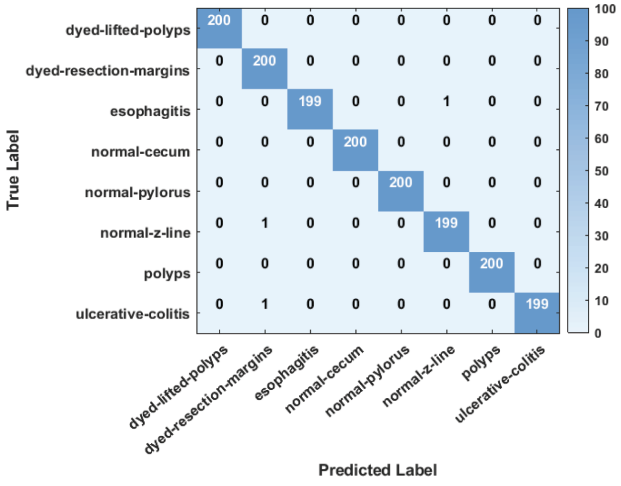
The strengths of the proposed ESOA with Stacked LSTM-SAF and the limitations of existing techniques are discussed in this section. Existing techniques face several drawbacks: (1). CSVM [16] struggled with poor scalability in large dataset. (2). LPNet [17] was sensitive to noise and texture variations because wavelet decomposition amplified high-frequency components, leading to noise overlap and misclassification. (3). ResNet18 [18] had relatively shallow depth, which limited its ability to capture highly complex features, resulting in inaccurate performance. (4). EDN [19] lost fine-grained spatial information due to encoding process. (5). SNet [20] had limited capacity to capture complex and subtle features due to shallow feature extraction layers, leading to suboptimal performance. The proposed ESOA with Stacked LSTM-SAF overcomes these limitations. ESOA ensures optimal feature selection by removing redundant features, enhancing data relevance and reducing training time. The Stacked LSTM-SAF captures both short-term and long-term dependencies, maintains stable training, avoids overfitting, and improves convergence efficiency. As a result, the proposed method achieves higher accuracy, reliability, and robustness compared with existing methods.

V. Conclusion

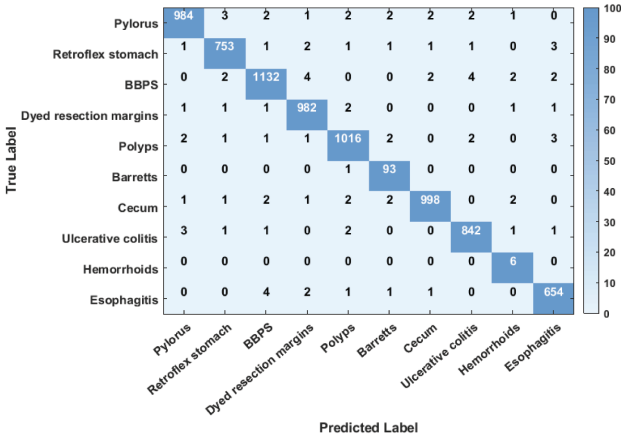
This research proposed ESOA with Stacked LSTM-SAF to accurately classify GI diseases. The proposed method achieved superior accuracies of 99.60% and



(A) KVASIR-V1

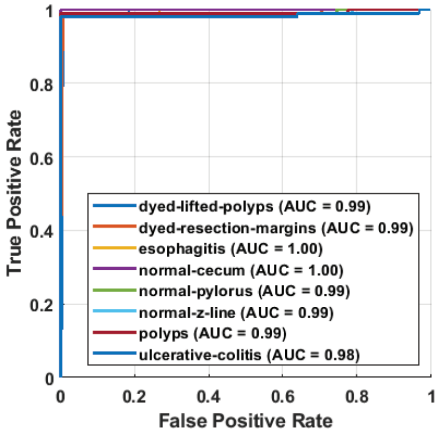


(B) Kvasir-V2

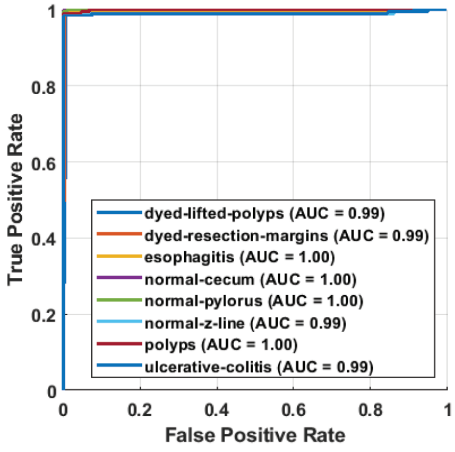


(C) HyperKvasir

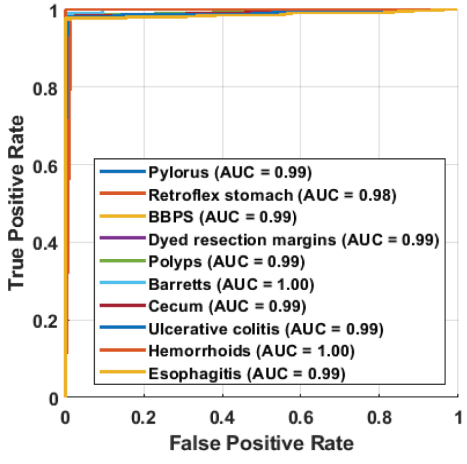
Figure 6: Evaluation of confusion matrix for Stacked LSTM-SAF **(A)** Kvasir-V1, **(B)** Kvasir-V2, **(C)** HyperKvasir. Stacked LSTM-SAF, stacked long short-term memory with Sech activation function.



(A) KVASIR-V1

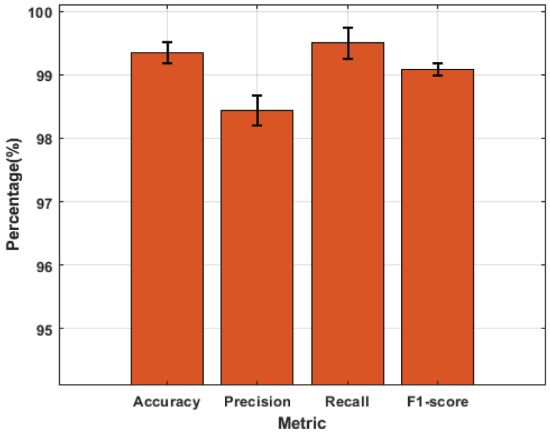


(B) Kvasir-V2

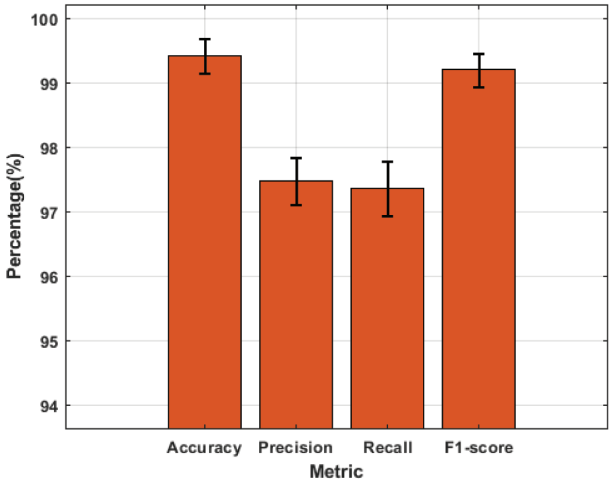


(C) HyperKvasir

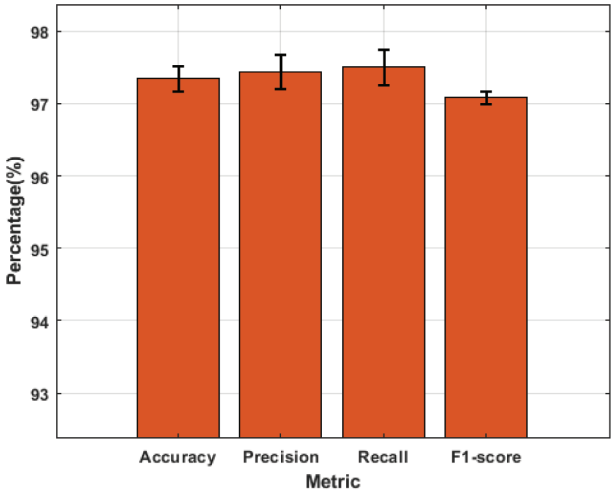
Figure 7: Evaluation of RoC curve for Stacked LSTM-SAF: **(A)** Kvasir-V1, **(B)** Kvasir-V2, **(C)** HyperKvasir. ROC, receiver operating characteristics; Stacked LSTM-SAF, stacked long short-term memory with Sech activation function.



(A) Kvasir-V1



(B) Kvasir-V2



(C) HyperKvasir

Figure 8: Analysis of standard deviation for the proposed method: **(A)** Kvasir-V1, **(B)** Kvasir-V2, **(C)** HyperKvasir.

Table 4: Performance analysis of computational complexity across datasets

Methods	Datasets	Memory consumption (MB)	Training time (s)	Inference time (s)
RNN	**Kvasir-V1**	27.12	37.95	36.74
GRU		23.05	32.58	30.84
LSTM		8.67	30.58	26.59
Stacked-LSTM-SAF		**6.72**	**2.54**	**9.58**
RNN	**Kvasir-V2**	27.48	38.62	37.12
GRU		22.87	35.12	31.04
LSTM		9.02	31.28	27.11
Stacked-LSTM-SAF		**6.58**	**2.73**	**9.92**
RNN	**HyperKvasir**	30.48	42.69	40.98
GRU		27.46	25.96	22.02
LSTM		21.69	19.78	17.36
Stacked-LSTM-SAF		**8.46**	**5.80**	**11.96**

GRU, gated recurrent unit; LSTM, long short-term memory; MB, RNN, recurrent neural network; Stacked LSTM-SAF, stacked long short-term memory with Sech activation function.

Table 5: Cross-dataset validation results: Trained on Kvasir-V1 and tested on Kvasir-V2

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
RNN	87.35	88.20	86.50	87.34
GRU	91.12	92.05	90.50	91.27
LSTM	94.50	95.10	93.80	94.44
Stacked LSTM-SAF	**97.25**	**96.80**	**96.00**	**96.39**

GRU, gated recurrent unit; LSTM, long short-term memory; RNN, recurrent neural network; Stacked LSTM-SAF, stacked long short-term memory with Sech activation function.

Table 6: Comparative analysis of existing methods on Kvasir-V1 and V2 datasets

Methods	Datasets	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
LPNet [17]	Kvasir-V1	93.55	93.55	93.55	93.55
VGG16 kernle RBF [19]	Kvasir-V2	96.64	97	97	97
SK-Net [20]	Kvasir-V1	98.45	N/A	96.60	N/A
	Kvasir-V2	97.83	N/A	N/A	N/A
Star-GAN + InceptionNet-V3 [21]	Kvasir-V2	94.96	N/A	94.93	94.93
CapsNet [22]	Kvasir-V2	93.40	N/A	N/A	N/A
Proposed ESOA with Stacked LSTM-SAF	**Kvasir-V1**	**99.60**	**98.71**	**99.88**	**99.20**
	Kvasir-V2	**99.88**	**97.93**	**97.12**	**99.61**

ESOA, enhanced skill optimization algorithm; Stacked LSTM-SAF, stacked long short-term memory with Sech activation function; Star-GAN, star-generative adversarial network.

99.88% on the Kvasir-V1 and V2 datasets, demonstrating enhanced diagnostic performance. These results indicate that integrating ESOA with Stacked LSTM-SAF improves convergence speed, generalization, and computational efficiency across the Kvasir-V1, Kvasir-V2, and HyperKvasir datasets. Practically, this method supports endoscopic image analysis by assisting clinicians in GI disease classification, reducing diagnostic errors, and improving patient outcomes. Compared with existing methods such as SK-Net, the proposed method achieves higher accuracy on both the Kvasir-V1 and V2 datasets. Furthermore, the method attains 99.74% accuracy on the the HyperKvasir dataset, demonstrating strong generalization on unseen data. However, the SAF limits feature diversity due to its output range being bounded between 0 and 1, which may reduce discriminative capability. Additionally, the ESOA with Stacked LSTM-SAF model lacks transparency, making it difficult to understand decision processes and build user trust. Therefore, Explainable Artificial Intelligence will be considered in future work to enhance interpretability and trustworthiness. Furthermore, class imbalance will be addressed in the preprocessing stage to further improve model performance.

References

[1] X. Qiao, C. Lu, M. Duan, Z. Liu, Y. Liu, W. Chen, R. Gao, Z. Li, and X. Zuo, "Intelligent tongue diagnosis model for gastrointestinal diseases based on tongue images," *Biomed. Signal Process. Control*, vol. 96, 106643, 2024

[2] M. Vania, B.A. Tama, H. Maulahela, and S. Lim, "Recent advances in applying machine learning and deep learning to detect upper gastrointestinal tract lesions." *IEEE Access*, 2023.

[3] S. Asif, M. Zhao, F. Tang, and Y. Zhu, "DCDS-Net: deep transfer network based on depth-wise separable convolution with residual connection for diagnosing gastrointestinal diseases," *Biomed. Signal Process. Control*, vol. 90, 105866, 2024.

[4] D. You, Y. Wu, M. Lu, F. Shao, Y. Tang, S. Liu, L. Liu, Z. Zhou, R. Zhang, S. Shen, and T. Lange, "A genome-wide cross-trait analysis characterizes the shared genetic architecture between lung and gastrointestinal diseases," *Nature Communications*, vol. 16, no. 1, 3032, 2025

[5] Z. Waheed, J. Gui, K. Amjad, I. Waheed, and S. Asif, "An ensemble approach of deep CNN models with beta normalization aggregation for gastrointestinal disease detection," *Biomed. Signal Process. Control*, vol. 105, 107567, 2025.

[6] A. A. Demirbaş, H. Üzen, and H. Fırat, "Spatial-attention ConvMixer architecture for classification and detection of gastrointestinal diseases using the Kvasir dataset." *Health Information Science and Systems*, vol. 12, no. 1, 32, 2024.

[7] M.A. Elmagzoub, S. Kaur, S. Gupta, A. Rajab, K. Rajab, M.S. Al Reshan, H. Alshahrani, and A. Shaikh, "Improving Endoscopic Image Analysis: Attention Mechanism Integration in Grid Search Fine-tuned Transfer Learning Model for Multi-class Gastrointestinal Disease Classification." *IEEE Access*, 2024.

[8] H.A. Pokuaa, A.F. Adekoya, B.A. Weyori, and O. Nyarko-Boateng, "Patch-and-Amplify Capsule Network for the Recognition of Gastrointestinal Diseases" *Scientific African,* e02277, 2024.

[9] A.S. Almasoud, M. Maray, A.A. Albraikan, F. Alotaibi, M.M. Alnfiai, and A. Sayed, "Gastrointestinal Cancer Detection and Classification Using African Vulture Optimization Algorithm with Transfer Learning". *IEEE Access*, 2024.

[10] O.M. Mirza, A. Alsobhi, T. Hasanin, M.K. Ishak, F.K. Karim, and S.M. Mostafa, "Computer Aided Diagnosis for Gastrointestinal Cancer Classification using Hybrid Rice Optimization with Deep Learning." *IEEE Access*, 2023.

[11] M. Patel, K. Gohil, A. Gohil, F. Ramoliya, R. Gupta, S. Tanwar, Z. Polkowski, F. Alqahtani, and A. Tolba, "Explainable AI for gastrointestinal disease diagnosis in telesurgery Healthcare 4.0." *Computers and Electrical Engineering*, vol. 118, 109414, 2024.

[12] A.H. Rabie, and A.I. Saleh, "Diseases diagnosis based on artificial intelligence and ensemble classification". *Artificial Intelligence in Medicine*, vol. 148, 102753, 2024.

[13] S. Asif, "A Fuzzy Minkowski Distance-based fusion of convolutional neural networks for gastrointestinal disease detection." *Applied Soft Computing*, vol. 158, 111595, 2024.

[14] D. Bajhaiya and S. N. Unni, "Deep learning-enabled detection and localization of gastrointestinal diseases using wireless-capsule endoscopic images," *Biomed. Signal Process. Control*, vol. 93, 106125, 2024.

[15] H. Guo, S.A. Somayajula, R. Hosseini, and P. Xie, "Improving image classification of gastrointestinal endoscopy using curriculum self-supervised learning." *Scientific Reports*, vol. 14, no. 1, 6100, 2024.

[16] F. Mohammad, and M. Al-Razgan, "Deep feature fusion and optimization-based approach for stomach disease classification." *Sensors*, vol. 22, no. 7, 2801, 2022.

[17] P. Sharma, D. Das, A. Gautam, and B. K. Balabantaray, "LPNet: A lightweight CNN with discrete

wavelet pooling strategies for colon polyp's classification," *Int. J. Imaging Syst. Technol.*, vol. 33, no. 2, 495–510, 2023.

[18] J. Naz, M. I. Sharif, M. I. Sharif, S. Sharif, H. T. Rauf, and A. E. Ragab, "A comparative analysis of optimization algorithms for gastrointestinal abnormalities recognition and classification based on ensemble XcepNet23 and ResNet18 features," *Biomedicines*, vol. 11, no. 6, 1723, 2023.

[19] D. H. Intissar and B. A. Yassine, "Detecting early gastrointestinal polyps in histology and endoscopy images using deep learning," *Front. Artif. Intell.*, vol. 8, 1571075, 2025.

[20] S. Siddiqui, J. A. Khan, T. Akram, M. Alharbi, J. Cha, and D. A. AlHammadi, "SNet: A novel convolutional neural network architecture for advanced endoscopic image classification of gastrointestinal disorders," *SLAS Technol.*, 100304, 2025.

[21] H. C. Park, I. P. Hong, S. Poudel, and C. Choi, "Data augmentation based on generative adversarial networks for endoscopic image classification," *IEEE Access*, vol. 11, 49216–49225, 2023.

[22] F. E. Oğuz and A. Alkan, "Weighted ensemble deep learning approach for classification of gastrointestinal diseases in colonoscopy images aided by explainable AI," *Displays*, vol. 85, 102874, 2024.

[23] Kvasir V1 dataset: https://datasets.simula.no/kvasir/ (Accessed on Jan 2025).

[24] Kvaris V2 dataset: https://www.kaggle.com/datasets/plhalvorsen/kvasir-v2-a-gastrointestinal-tract-dataset (Accessed on Jan 2025).

[25] Hayum, J. Jaya, R. Sivakumar, and B. Paulchamy, "An efficient breast cancer classification model using bilateral filtering and fuzzy convolutional neural network," *Sci. Rep.*, vol. 14, no. 1, 6290, 2024.

[26] Sharma and R. Parvathi, "Enhancing cervical cancer classification: Through a hybrid deep learning approach integrating DenseNet201 and InceptionV3," *IEEE Access*, 2025, doi: 10.1109/ACCESS. [If DOI available, I can update].

[27] N. Alshdaifat, H. A. Owida, Z. Mustafa, A. Aburomman, S. Abuowaida, A. Ibrahim, and W. Alsharafat, "Automated blood cancer detection models based on EfficientNet-b3 architecture and transfer learning," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 36, no. 3, 1731–1738, 2024.

[28] H. Givi and M. Hubalovska, "Skill optimization algorithm: a new human-based metaheuristic technique," *Comput. Mater. Contin.*, vol. 74, no. 1, 179–202, 2023.

[29] M. Sharaf, E.E.D. Hemdan, A. El-Sayed and NA. El-Bahnasawy, "An efficient hybrid stock trend prediction system during COVID-19 pandemic based on stacked-LSTM and news sentiment analysis." *Multimedia Tools and Applications*, vol. 82, no. 16, 23945-23977, 2023.