BMC Medical Informatics and Decision Making

RESEARCH

Open Access

A novel network-level fused deep learning architecture with shallow neural network classifier for gastrointestinal cancer classification from wireless capsule endoscopy images

Muhammad Attique Khan[1*], Usama Shafiq[2], Ameer Hamza[3], Anwar M. Mirza[1], Jamel Baili[4], Dina Abdulaziz AlHammadi[5], Hee-Chan Cho[6] and Byoungchol Chang[7*]

Abstract

Deep learning has significantly contributed to medical imaging and computer-aided diagnosis (CAD), providing accurate disease classification and diagnosis. However, challenges such as inter- and intra-class similarities, class imbalance, and computational inefficiencies due to numerous hyperparameters persist. This study aims to address these challenges by presenting a novel deep-learning framework for classifying and localizing gastrointestinal (GI) diseases from wireless capsule endoscopy (WCE) images. The proposed framework begins with dataset augmentation to enhance training robustness. Two novel architectures, Sparse Convolutional DenseNet201 with Self-Attention (SC-DSAN) and CNN-GRU, are fused at the network level using a depth concatenation layer, avoiding the computational costs of feature-level fusion. Bayesian Optimization (BO) is employed for dynamic hyperparameter tuning, and an Entropy-controlled Marine Predators Algorithm (EMPA) selects optimal features. These features are classified using a Shallow Wide Neural Network (SWNN) and traditional classifiers. Experimental evaluations on the Kvasir-V1 and Kvasir-V2 datasets demonstrate superior performance, achieving accuracies of 99.60% and 95.10%, respectively. The proposed framework offers improved accuracy, precision, and computational efficiency compared to state-of-the-art models. The proposed framework addresses key challenges in GI disease diagnosis, demonstrating its potential for accurate and efficient clinical applications. Future work will explore its adaptability to additional datasets and optimize its computational complexity for broader deployment.

Keywords Gastrointestinal disease, Wireless capsule endoscopy, Deep learning, LSTM, Fusion, Optimization, Shallow machine learning

*Correspondence:
Muhammad Attique Khan
attique.khan@ieee.org
Byoungchol Chang
bcchang@hanyang.ac.kr

Full list of author information is available at the end of the article

Introduction

Gastrointestinal (GI) diseases are among the most prevalent global health concerns, affecting millions annually [1]. These conditions, including ulcers, polyps, bleeding esophagitis, and colorectal cancer, can lead to severe health complications and significantly impact the quality of life if not diagnosed and treated early [2, 3]. For instance, studies indicate that in the United States alone, approximately 27,510 new cases of stomach-related illnesses are reported annually, with 11,140 fatalities [4, 5]. Globally, colorectal cancer claims the lives of an estimated 694,000 individuals annually, particularly in developing countries where healthcare access remains limited [6, 7]. Early diagnosis is critical, as it increases survival rates by up to 80% in many cases, highlighting the importance of efficient and accurate diagnostic systems [8, 9].

Wireless Capsule Endoscopy (WCE) has emerged as a transformative technology in gastroenterology, enabling non-invasive, high-resolution imaging of the GI tract [10, 11]. A small capsule equipped with a camera is ingested by the patient, recording video frames as it traverses the digestive system. These frames are transmitted to an external recorder, providing clinicians with a comprehensive view of the GI tract [12, 13]. While WCE has revolutionized diagnostic workflows, manually analyzing the vast number of images captured is time-intensive, highly subjective, and prone to human error [14]. Even skilled physicians find it challenging to accurately identify diseased regions amidst the substantial visual variability in GI images [15].

Recently developed automated computer-aided systems have proven very helpful for physicians in their clinical work for several applications such as stomach cancer, brain tumor, skin cancer [16], and many more [17–19]. Most computer vision researchers have created automated diagnosis systems that accurately detect and classify GIT cancer [20]. These methods are based on supervised learning algorithms [21]. These automated diagnosis systems get the right information by using endoscopic video frames and relevant traits that can help find gastrointestinal tract (GIT) cancer automatically [22]. To automatically identify gastrointestinal tract diseases, numerous computer-aided algorithms extracted handmade elements such as texture, color, and shape [23]. To categorize GI cancer, several studies extracted information using the discrete wavelet transform and the discrete cosine transform. Moreover, color characteristics function as an additional kind of descriptor for analysis by assisting in the extraction of color-based information from WCE photos that have been infected [24]. Computer-aided diagnosis (CAD), which previously relied on conventional features and convolutional neural networks to extract high-level information from WEC pictures and improve CAD system performance, has recently revolutionized because of deep learning. Developing deep learning raises the accuracy of cancer diagnosis and recognition [25]. CAD systems extract features from GIT endoscopic images to diagnose the condition. However, not all features are useful, and features need to be changed from higher-dimensional to lower-dimensional by using a feature selection algorithm to get rid of features that aren't needed. Optimizing feature selection aids in choosing pertinent features and improves CAD performance as a whole [26]. The optimal subset of feature vectors from the original feature vector can be chosen using a variety of optimization procedures, including genetic algorithms (GA), marine predator optimization, entropy selection, gray wolf optimization, and Bayesian optimization [27, 28]. The methods performed satisfactorily in terms of accuracy and calculation time in this study. The comprehensive feature selection process used MPA, which improved the outcomes even more. Several CV researchers have demonstrated the significance of the MPA (Marine Predators Algorithm) in addressing dimensionality reduction and feature selection issues [29]. Automating the detection and diagnosis of gastrointestinal disorders with wireless capsule endoscopy (WCE) images has been the subject of extensive research in recent years [26, 30]. Machine learning and deep learning methods are used to identify disease regions accurately and effectively [31]. The main conclusions and contributions of earlier research in this area were summarized in the overview. Kalinin et al. [32] presented two deep convolutional neural network applications for medical image segmentation. The first application used a wireless capsule endoscope to identify gastrointestinal bleeding by distinguishing between angiodysplasia lesions in films. By using different deep architectures with ImageNet pretrained encoders, performance can be improved. Compared to the U-Net architecture, the second application, which involved segmenting surgical tools in robotic surgical footage, produced competitive results.

To address these challenges, computer-aided diagnosis (CAD) systems leveraging medical image processing have become increasingly prevalent. Early CAD systems primarily relied on handcrafted features, such as texture, color, and shape, for disease classification. However, these methods were often limited by their inability to generalize effectively across diverse datasets and disease types. Recent advancements in deep learning, particularly Convolutional Neural Networks (CNNs), have significantly improved feature extraction, offering state-of-the-art accuracy and reliability in medical image analysis. Despite these advancements, key challenges remain:

- **Class imbalance**: Most publicly available datasets exhibit significant class imbalances, where healthy samples outnumber diseased cases, negatively

affecting model training and classification accuracy [27, 28].

- **Inter- and intra-class similarities**: GI images often display substantial visual overlap between healthy and diseased regions or among different disease categories, complicating classification task. A visual example can be shown in Fig. 1.
- **Computational complexity**: Many existing models rely on deep architectures with a high number of parameters, leading to increased computational requirements and training times.

The proposed network-level fusion addresses some main challenges in classifying gastrointestinal diseases. One of the most critical challenges faced is class imbalance, which often prevents the performance of the deep learning model due to the dominance of healthy samples over diseased cases. The study addresses this problem through a comprehensive data augmentation pipeline to ensure better generalization. Furthermore, the architecture addresses Inter- and Intra-class similarities, a persistent problem in the gastrointestinal dataset that shows a significant visual overlap between healthy and sick regions and different disease categories. Incorporating self-attention dynamically focuses on disease-related areas, improving the discrimination of features despite these visual complexities. Finally, the computational complexity is effectively reduced by replacing the traditional convolutional layer with sparse convolutional operations, reducing the number of parameters and memory requirements and preserving the efficiency of feature extraction. Experimental results on the Kvasir-V1 and Kvasir-V2 datasets demonstrate the framework's efficacy, achieving superior accuracy and computational efficiency compared to state-of-the-art (SOTA) models. Furthermore, GradCAM-based visualizations enhance interpretability, offering clinician's valuable insights into the decision-making process. This study proposes a novel deep-learning framework to overcome these limitations and enhance GI disease classification from WCE images. Key contributions of this work include:

- **Data augmentation**: To address class imbalance and improve model generalization, a comprehensive augmentation pipeline is employed, increasing data diversity and robustness.
- **Sparse convolutional denseNet201 with self-attention (SC-DSAN)**: A modified version of DenseNet201 with sparse convolutional layers reduces computational complexity, while a self-attention mechanism improves feature extraction.
- **CNN-GRU hybrid architecture**: The integration of Convolutional Neural Networks (CNNs) with Gated Recurrent Units (GRUs) leverages spatial and sequential dependencies, enabling more accurate disease classification.

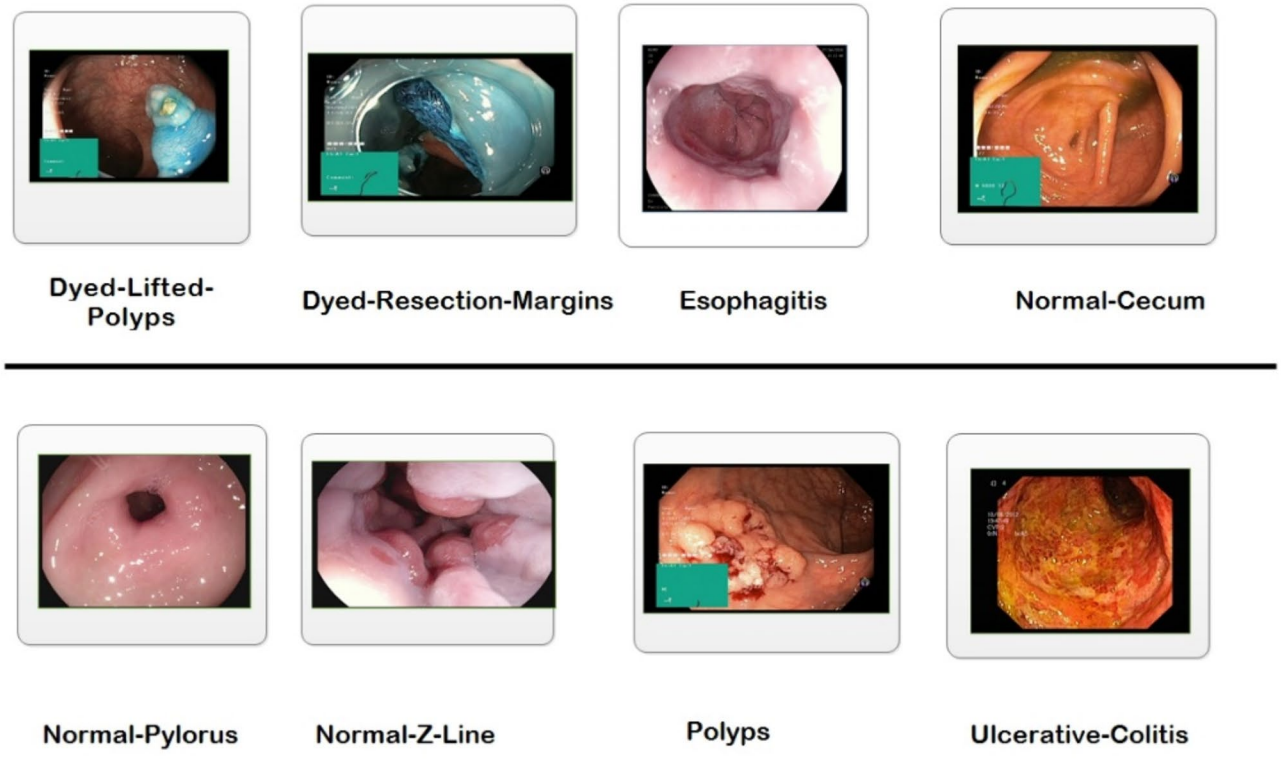

Fig. 1 Samples of the Kavsir dataset [33]

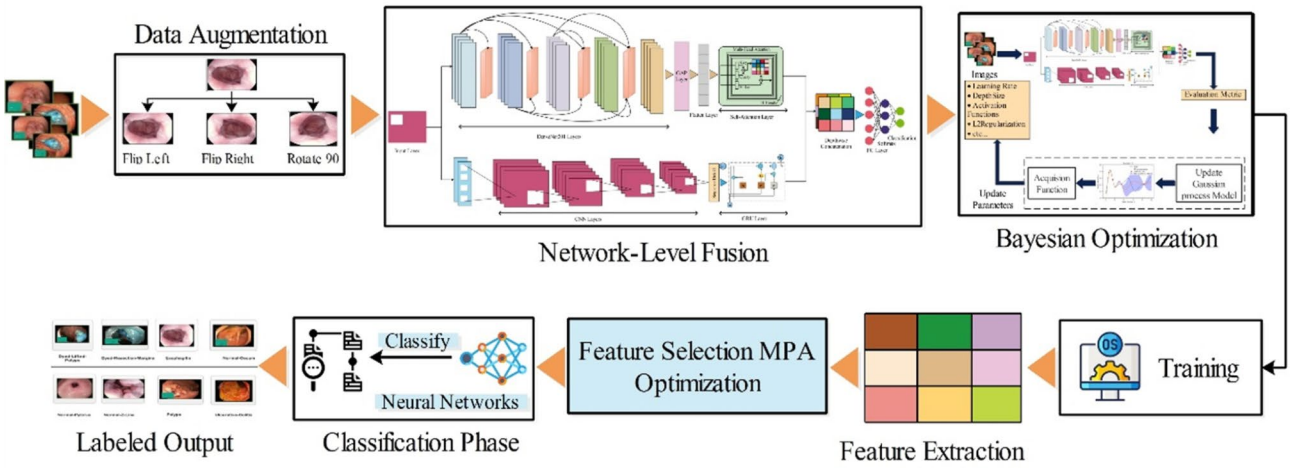

Fig. 2 The proposed architecture of Kvasir database classification by using modified DenseNet-201 and CNN-GRU

- **Network-level fusion**: A novel depth concatenation layer fuses SC-DSAN and CNN-GRU architectures, avoiding the resource-intensive nature of feature-level fusion.
- **Optimization techniques**: Bayesian Optimization (BO) dynamically tunes hyperparameters and the Entropy-controlled Marine Predators Algorithm (EMPA) selects optimal features, enhancing performance.

The remainder of this paper is organized as follows: Sect. "Proposed Methodology" describes the methodology, including data preparation, model architectures, and optimization techniques. Section "Results and Discussion" presents experimental results and analyses. Section "Discussion" discusses the findings and compares the proposed framework with SOTA methods. Finally, Section "Conclusion" concludes the study with recommendations for future research.

Proposed methodologyss

The proposed automatic network-level fusion framework is illustrated in Fig. 2. The process begins with data augmentation, which enhances the diversity of the dataset and improves the robustness of the model during training. Following this, two architectures—Sparse Convolutional DenseNet201 with Self-Attention (SC-DSAN) and a customized CNN-GRU—are integrated using a depthwise concatenation layer. This fusion leverages the strengths of both architectures to enhance the model's learning capabilities. The fused model is then trained on the selected dataset, and deep features are extracted from the depthwise activation layer. To optimize feature selection, the Marine Predators Algorithm (MPA) is applied, ensuring that only the most relevant features are utilized. Finally, the optimized features are classified using six different classifiers: Narrow Neural Network (NNN), Medium Neural Network (MNN), Wide Neural Network (WNN), Bi-layered Neural Network (BNN), Tri-layered Neural Network (TNN), and cubic SVM.

Table 1 The brief description of selected Kvasir V1 and VII datasets

Classes	Original image(V1/V2)	Augmented images	Training/ Testing image
Ulcerative Colitis	500/1000	4000	2000/2000
Normal-Pylorus	500/1000	4000	2000/2000
Normal-Cecum	500/1000	4000	2000/2000
Normal-z-line	500/1000	4000	2000/2000
Esophagitis	500/1000	4000	2000/2000
Dyed-Resection- Margins	500/1000	4000	2000/2000
Dyed-L-Polyps	500/1000	4000	2000/2000
Polyps	500/1000	4000	2000/2000

Dataset collection and normalizations

Our study utilizes the KVASIR database [33], a well-curated public resource for gastrointestinal (GI) analysis. This dataset includes eight annotated classes representing various GI tract conditions, with images depicting endoscopic procedures, pathological features, and anatomical landmarks. The KVASIR database is highly suitable for tasks involving image retrieval, machine learning, deep learning, and transfer learning, providing labeled samples for each class. However, special attention is required when utilizing images containing green hues in the frames, as they may impact the accuracy of endoscopic findings. A detailed description of the selected Kvasir datasets is presented in Table 1, with sample images shown in Fig. 1.

Sparse convolutional denseNet201 with self-attention (SC-DSAN)ss

DenseNet201, a modified variant of the original DenseNet, is specifically designed to enhance image classification tasks through its unique architecture. In DenseNet201, each layer receives feature maps from all preceding layers and passes its own feature maps to all subsequent layers [34]. This dense connectivity optimizes information flow, ensuring efficient feature reuse throughout the network, which leads to improved learning and overall performance. Compared to traditional networks such as ResNet, DenseNet201 is more parameter-efficient because it reduces the need for additional parameters by reusing features across layers. The intricate interconnections between layers help mitigate the vanishing gradient problem, allowing gradients to propagate smoothly across the network during backpropagation. This feature improves gradient flow, enhancing training stability, especially for very deep networks. Consequently, DenseNet201 boosts both accuracy and efficiency, making it a highly effective architecture for applications involving complex visual data [34, 35]. Its dense connections, feature reuse, and efficient gradient propagation make DenseNet201 a standout architecture in modern computer vision tasks within deep learning.

In this work, DenseNet201 is modified by replacing all traditional convolutional layers with Sparse Convolutional Layers (SCL) to reduce parameter redundancy through sparse decomposition. The optimal sparsity is achieved by exploiting both intra-channel and inter-channel redundancies. This is followed by a fine-tuning phase that mitigates any potential loss of feature identification resulting from the increased sparsity. Next, a flattening layer is introduced to convert the 2D feature maps into a 1D vector. Subsequently, a multi-headed self-attention layer is added to enhance feature representation. At the final stage, fully connected layers, a softmax layer, and a classification layer are introduced to complete the network architecture. The structure of the proposed SC-DSAN is illustrated in Fig. 3. Once the model is adjusted, it is trained on the selected Kvasir datasets, and the features are extracted from the self-attention activation, with the extracted feature dimension being N x 1920.

Novelty: sparse convolutional operations

Let the input feature vector be $\bar{F}$ in $\mathbb{R}^{S_h \times S_w \times D}$, where S_h, S_w and D represent the height, width, and depth of the feature vector, respectively. Also, the convolutional filter be ϕ in $\mathbb{R}^{k \times k \times D \times z}$, where k is the size of the filter and z is the resultant number of channels. We assume that the convolution is performed with zero padding and stride 1. Then, the output feature vector maps $\psi \in \mathbb{R}^{(S_h-k+1)\times (S_w-k+1)\times z} = \phi \times \bar{F}$ resulting from the convolutional layer are given by Eq. (1).

$$\psi(\alpha, \beta, \gamma) = \sum_{a=1}^{D} \sum_{i,j=1}^{k} \phi(i, j, a, \gamma) \bar{F}(\alpha + i - 1, \beta + j - 1, a) \quad (1)$$

The goal is to replace computationally expensive convolutional operation $\psi = \phi \times \bar{F}$ with the fast sparsified form, which is based on sparse matrix multiplication. To implement this process, we transform the tensor $\bar{F}$ into $\bar{\mathrm{P}} \in \mathbb{R}^{S_h \times S_w \times D}$ and filter ϕ into $\Psi \in \mathbb{R}^{k \times k \times D \times z}$ utilizing a matrix $\mathrm{M} \in \mathbb{R}^{D \times D}$ obtaining $\psi \approx \Psi \times \bar{\mathrm{P}}$ which defined by Eqs. (2–3):

$$\phi(i, j, a, \gamma) \approx \sum_{t=1}^{D} \Psi(i, j, a, \gamma) \mathrm{M}(\mathrm{t}, \mathrm{a}) \quad (2)$$

$$\bar{\mathrm{P}}(\alpha, \beta, a) = \sum_{t=1}^{D} \mathrm{M}(\mathrm{a}, \mathrm{t}) \bar{F}(\alpha, \beta, t) \quad (3)$$

For each channel $a = 1,2,\ldots D$, we decompose the tensor $\Psi(\cdot, \cdot, \mathrm{a}, \cdot) \in \mathbb{R}^{k \times k \times z}$ into the product of a matrix $\tau_i \in \mathbb{R}^{B_i \times z}$ and a tensor $\omega_i \in \mathbb{R}^{k \times k \times B_i}$ where B_i is the number of bases:

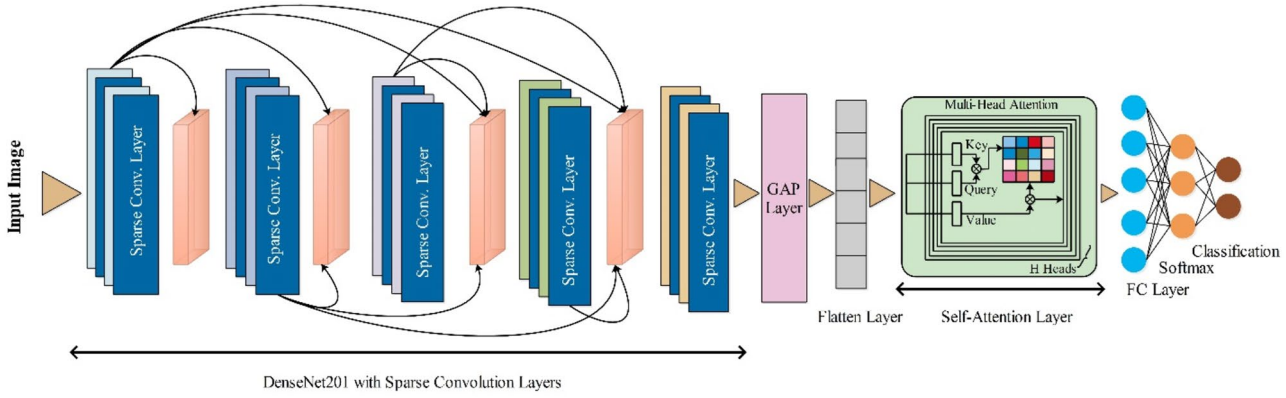

Fig. 3 High-level architecture of SC-DSAN with self-attention layer

$$\Psi\ (i,j,a,\gamma\ ) \approx \sum {}^{B_i}_{k=1}\tau_{\ i}\,(k,\gamma\ )\,\omega_{\ i}\,(i,j,k) \quad (4)$$

$$\partial_{\ i}\,(\alpha\ ,\beta\ ,k) = \sum {}^{k}_{i,j=1}\omega_{\ i}\,(i,j,k)\ \bar{\mathrm{P}}\ (\alpha\ +i,\beta\ +j-1,i) \quad (5)$$

Thus the approximated filter ϕ can be expressed by Eq. (6):

$$\phi\,(\alpha\ ,\beta\ ,k) \approx \sum {}^{D}_{i=1} \sum {}^{B_i}_{k=1}\tau_{\ i}\,(k,\gamma\ )\,\partial_{\ i}\,(\alpha\ ,\beta\ ,k) \quad (6)$$

Finally, the tensors ϕ and $\partial_{\ i}$ are combined by concatenating $\tau_{\ i}$ and $\partial_{\ i}$ along the dimension B_i. This results in the proposed sparse convolutional kernel Ψ which produces an output vector that closely approximate the original convolutional kernel ϕ. The sparse convolution operation is visually presented in Fig. 4. The integration of Sparse Convolutional DenseNet201 and Self-Attention is an important advantage in the classification of gastrointestinal diseases. The architecture effectively reduces computational complexity and memory requirements by incorporating reduced convolution operations and allowing the processing of high-resolution endoscopic images to be faster and more efficient. This efficiency is achieved without compromising the extraction of features, as dense connectivity of the DenseNet201 ensures optimal recycle of information and stable gradient transmission for learning. Moreover, Self-attention mechanisms enhance the extraction of features by dynamically focusing on the refined variations that can be covered by changes in the lighting and texture of tissues. These innovations made SC-DSAN a powerful and accurate solution for the classification of gastrointestinal diseases.

Proposed CNN-GRU architecture

The CNN-GRU model combines the strengths of Convolutional Neural Networks (CNNs) and Gated Recurrent Units (GRUs) [36] to efficiently process data with both spatial and temporal dependencies. CNN layers extract spatial features using convolutional filters and pooling operations to highlight essential patterns while reducing dimensionality. These spatial features are then flattened and fed into GRU layers, which are particularly effective at capturing temporal dependencies and sequential patterns in the data. The GRU layer is employed instead of LSTM because GRU uses a simpler architecture, fewer gates and parameters. It facilitating faster training and reducing memory usage. Although LSTM is more powerful for capturing long-term dependency, GRU usually performs similar functions, making it an effective alternative that provides faster convergence and fewer overfits. Reduced complexity is particularly useful in cases where model simplicity and training speed are given priority without sacrificing significant performance.

This combination enables the CNN-GRU model to comprehensively analyze complex datasets, making it a versatile and powerful tool for various applications. In this work, the CNN-GRU model is designed with 25 convolutional layers using 1×1 and 3×3 filters, 2×2 strides, max pooling, and ReLU activations. A GRU layer with 1280 units is appended to the network to process the sequential features. The architecture concludes with a fully connected layer, a softmax layer, and a classification layer. GRUs, being simplified versions of Long Short-Term Memory (LSTM) units, offer lower computational cost and faster convergence. Unlike LSTMs, GRUs utilize only two gates: the reset gate (R_t) and the update gate (U_t). The GRU is mathematically defined by Eqs. (7–10):

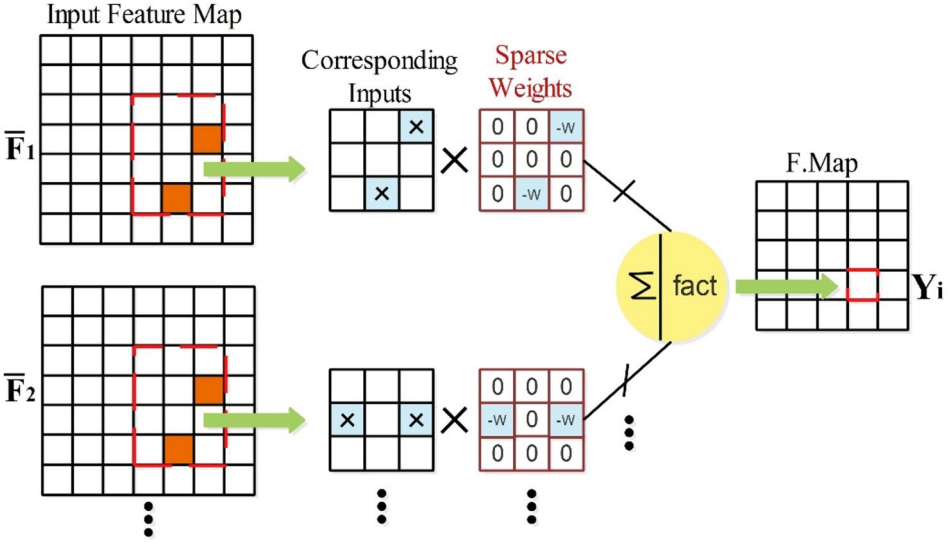

Fig. 4 Proposed sparse convolutional operation

$$R_t = sigmoid\left(\omega_R.[\Psi_{t-1}, I_t]\right) \tag{7}$$

$$U_t = sigmoid\left(\omega_U.[\Psi_{t-1}, I_t]\right) \tag{8}$$

$$\Psi_t = tanh\left(\omega.[R_t \odot \Psi_{t-1}, I_t]\right) \tag{9}$$

$$\mathrm{t}: \Psi_t = (1 - U_t) \odot \Psi_{t-1} + U_t \odot \Psi_t \tag{10}$$

Here I_t the input in time t. R_t presented the reset gate. U_t is the update gate, Ψ_t is the new memory and $\mathrm{t}: \Psi_t$ is the final memory state. The operator $\odot$ denotes element-wise multiplication. Once designed, the CNN-GRU model is trained on the selected datasets. Features are extracted from the GRU activation layer, yielding a feature vector with dimensions $N \times 1280$. The architecture of the proposed CNN-GRU network is visually represented in Fig. 5.

The SC-DSAN proposed a new gastrointestinal disease classification architecture in the Kvasir v1 and Kvasir v2. With the integration of small-scale convergence operations, SC-DSAN reduces computational complexity and memory requirements and preserves detailed representations of functions critical to processing high-resolution endoscopic images. The dense connectivity in DenseNet201 is further enhanced by the self-attention mechanism that dynamically focuses on the regions related to the disease. This capability is essential for different and complex samples, such as Kvasir v1 and Kvasir v2, where various lighting and tissue textures may mask disease patterns. Meanwhile, the CNN-GRU captures the temporal information and spatial relationship inherent in gastrointestinal disease data from Kvasir v1 and Kvasir v2. This fusion enables the model to learn fine spatial information and contextual relationships, thereby improving the classification performance in situations where the features of the disease can be stated within multiple frames. The integration of CNN-GRU not only enhances the reliability of the classification task but also enables greater generalization of the complex and diverse conditions found in Kvasir's dataset, finally establishing a new standard for the classification of gastrointestinal diseases.

Network level fusion and trainings

In this section, the proposed networks are fused using a depth concatenation layer, which is more efficient and less resource-intensive than feature-level fusion. Feature-level fusion typically requires significant computational resources and is time-consuming, whereas network-level fusion via depth concatenation combines the strengths of both architectures more effectively. Specifically, the depth concatenation layer merges the channels of the CNN-GRU model and the SC-DSAN model. This is followed by fully connected layers and a softmax layer to produce the final classification outputs, as illustrated in Fig. 6.

After fusion, the resulting model is trained on the selected datasets. Bayesian Optimization (BO) is employed during the training process to initialize and tune hyperparameters dynamically, ensuring optimal model performance. The BO algorithm used for this purpose is detailed in Sect. "Bayesian Optimization (BO)". For training and testing, the datasets are split into two equal portions, with 50% of the data used for training and the remaining 50% reserved for testing.

Bayesian optimization (BO)

Bayesian Optimization [37] is a highly efficient technique for hyperparameter tuning, particularly in complex architectures like Convolutional Neural Networks (CNNs). By employing a statistical model of the objective function, Bayesian Optimization efficiently identifies the optimal hyperparameters for evaluation, striking a balance between exploring new possibilities and exploiting existing knowledge [37].

The core concept behind Bayesian Optimization lies in the creation of a surrogate model, represented as $p(f|D)$, where the observed data D consists of hyperparameters and their corresponding loss values. A Gaussian Process (GP) is frequently used as the surrogate model and is mathematically defined as:

$$f(\phi)\left(m(\phi), \vartheta(\phi, \phi\prime)\right) \tag{11}$$

Here $m(\varphi)$ represents the mean function and $\vartheta(\varphi, \varphi\prime)$ is the kernel function that encodes the covariance among points. The posterior distribution of the

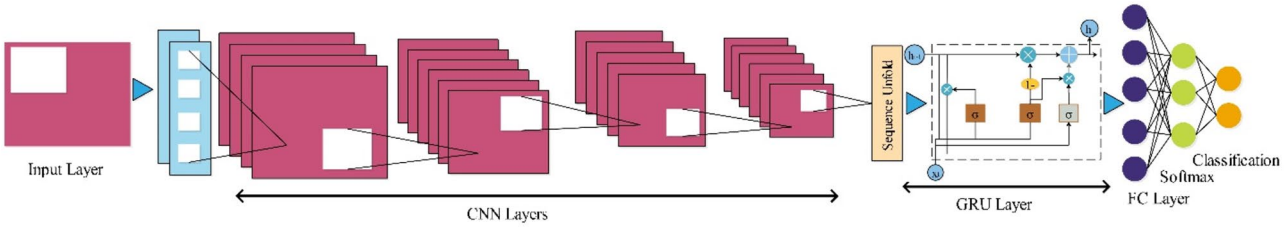

Fig. 5 Proposed architecture of CNN-GRU model

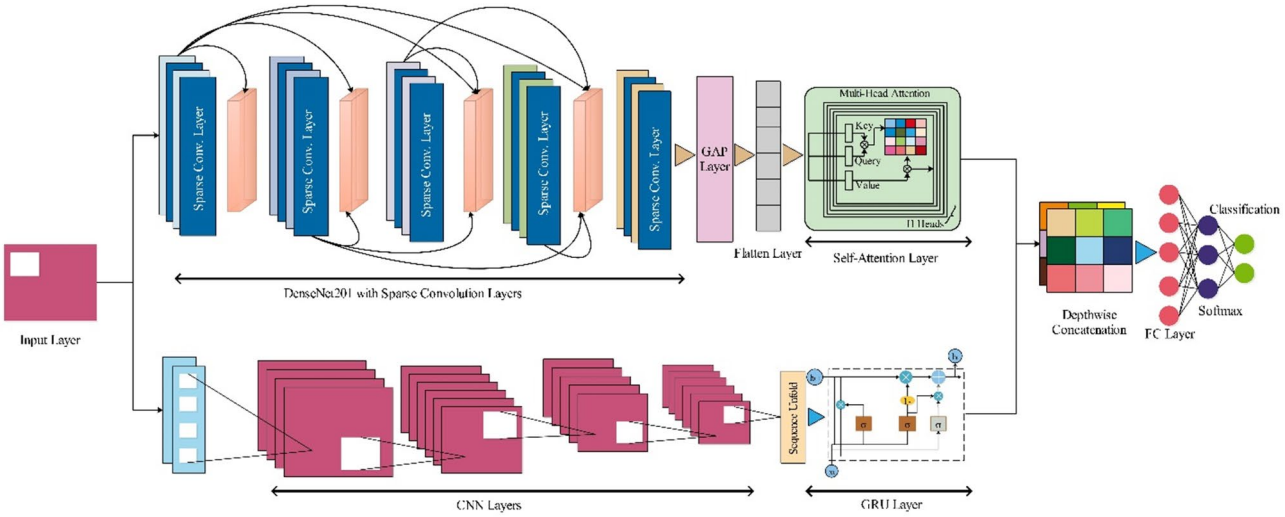

Fig. 6 Architecture of proposed network-level fusionss

Table 2 List of hyperparameters and their ranges for this work

HyperParameters	Ranges
Section Depth	[1, 4]
Momentum	[0.4, 0.98]
Dropout	[0.0, 0.046]
Activations	RELU, Clipped ReLU, Sigmoid
Learning Rate	[0.0021,0.96]
L2Regularization	$[1\,e^{-6}, 1e^{-1}]$

objective function is continuously updated in real time as new observations are collected, improving the surrogate model's accuracy.

The next set of hyperparameters to evaluate is determined by maximizing the acquisition function $\eta\ (\varphi \mid D)$. This function balances exploration (sampling new promising regions) and exploitation (refining known regions). Two commonly used acquisition functions are Upper Confidence Bound (UCB) and Expected Improvement (EI), defined as:

$$\psi_{EI} = \mathbb{E}[\max(0, f(\varphi) - f(\varphi^{+}))] \tag{12}$$

$$\psi_{UCB}(\varphi) = mean(\varphi) + \tau\,\partial\,(\varphi) \tag{13}$$

In these equations, $mean(\varphi), \tau\,\partial\,(\varphi)$ presented the mean and standard deviation estimated by the GP, φ^{+} is the best observed hyperparameter value, and τ is the balancing parameter that controls the trade-off between exploration and exploitation. In this study, Bayesian Optimization was employed to determine the optimal hyperparameters for training the proposed models. For the training, the epochs is 200. The specific hyperparameters considered for optimization are detailed in Table 2. The training curve for the KVASIR-V2 dataset is also illustrated in Fig. 7.

Testing model

After training the proposed models, 50% of the dataset is used for feature extraction. Prominent features are extracted from the self-attention layer of the modified DenseNet201, resulting in a feature dimension of $N \times 1920$. Simultaneously, features are extracted from the GRU activation layer of the CNN-GRU model, yielding a feature dimension of $N \times 1280$. To further refine the extracted features, the Entropy-Controlled Marine Predators Algorithm (EMPA) is applied. This optimization ensures that only the most relevant features are retained, enhancing the overall performance and efficiency of the classification process.

Entropy-controlled marine predators algorithm (MPA)

The Marine Predators Algorithm (MPA) [29] is a nature-inspired optimization technique modeled after the foraging behaviors of ocean predators such as sharks and whales. Its key attributes—being derivative-free and user-friendly—make it highly effective for a broad range of optimization problems. The MPA effectively balances exploration and exploitation in the search space by incorporating Levy and Brownian movements. These strategies mimic how marine predators navigate complex ecosystems, overcoming challenges such as currents and obstacles to locate prey efficiently.

The algorithm operates through three primary phases, each corresponding to different velocity scenarios observed in predator-prey interactions. By simulating these behaviors, the MPA refines candidate solutions and adapts dynamically to the optimization landscape, making it a robust tool for addressing complex problems [29].

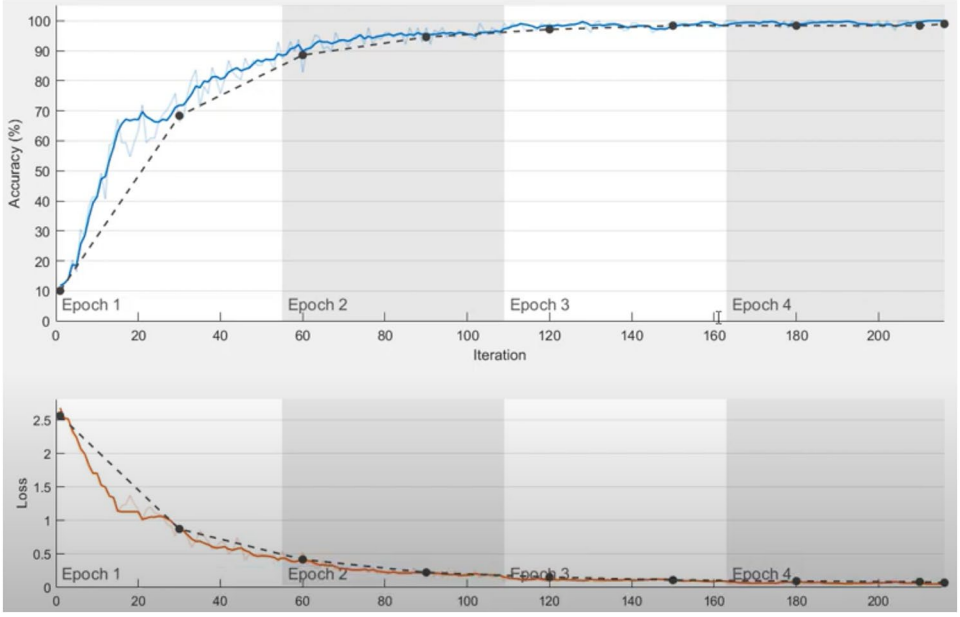

Fig. 7 Training plot of the proposed fused architecture on KVASIR-V2 dataset

$$Prey = \begin{bmatrix} A_{1,1} & A_{1,2} & \cdots & A_{1,d} \\ A_{2,1} & A_{1,2} & \cdots & A_{2,d} \\ \cdots & \cdots & \cdots & \cdots \\ A_{n,1} & A_{n,2} & \cdots & A_{n,d} \end{bmatrix} \quad (14)$$

$$Elite = \begin{bmatrix} B^I_{1,1} & B^I_{1,2} & \cdots & B^I_{1,d} \\ B^I_{2,1} & B^I_{1,2} & \cdots & B^I_{2,d} \\ \cdots & \cdots & \cdots & \cdots \\ B^I_{n,1} & B^I_{n,2} & \cdots & B^I_{n,d} \end{bmatrix} \quad (15)$$

Here n is the number of prey and d the number of variables. The Marine Predators Algorithm operates in three main phases

High-velocity phase

In this phase, the speed of the marine predator is higher than that of the prey. The predators stay stationary, not changing their location, while the prey may move in a Levy or Brownian pattern.

$$\overrightarrow{S_j} = \overrightarrow{Y_B}\left(\overrightarrow{Elite_j} - \overrightarrow{Y_B}\cdot\overrightarrow{Prey_j}\right),\ j = 1,2,\cdots,n \quad (16)$$

$$\overrightarrow{Prey_j} = \overrightarrow{Prey_j} + X\cdot\overrightarrow{Y}\cdot\overrightarrow{S_j},\ j = 1,2,\cdots,n \quad (17)$$

where $\overrightarrow{Y_B}$ holds random value matrix-holds from the Brownian motion. Where $X = 0.5$ but can be turned, the $\overrightarrow{Y}$ holds random values between 0 and 1.

Same-velocity phase

The speed of the marine predator becomes identical to that of the prey. In this scenario, when the prey moves in a Levy pattern, the most effective strategy for the predator is to adopt a Brownian motion.

$$\overrightarrow{S_j} = \overrightarrow{Y_L}\cdot\left(\overrightarrow{Elite_j} - \overrightarrow{Y_L}\cdot\overrightarrow{Prey_j}\right),\ j = 1,2,\cdots,n/2 \quad (18)$$

Here $\overrightarrow{Y_L}$ holds the random value matrix from the Levy distribution,

$$\overrightarrow{Prey_j} = \overrightarrow{Prey_j} + X\cdot\overrightarrow{Y}\cdot\overrightarrow{S_j} \quad (19)$$

Here $X = 0.5$ but can be turned, the $\overrightarrow{Y}$ holds random values between 0 and 1.

$$\overrightarrow{S_j} = \overrightarrow{Y_B}\cdot\left(\overrightarrow{Y_B}\cdot\overrightarrow{Elite_j} - \overrightarrow{Prey_j}\right),\ j = n/2+1, n/2+2,\cdots,n \quad (14)$$

$$\overrightarrow{Prey_j} = \overrightarrow{Prey_j} + Q\cdot XP\cdot\overrightarrow{S} \quad (20)$$

where $Q = 0.5$ but can be turned, $XP = \left(1 - \frac{t}{T}\right)2^{\frac{t}{T}}$ holds random values.

Low-velocity phase

In this phase, the speed of the marine predator is less than that of the prey. Here, the Levy motion is the best strategy for the predator to adopt.

$$\overrightarrow{S_j} = \overrightarrow{Y_L}\cdot\left(\overrightarrow{Y_L}\cdot\overrightarrow{Elite_j} - \overrightarrow{Prey_j}\right),\ j = 1,2,\cdots,n \quad (21)$$

$$\overrightarrow{Prey_j} = \overrightarrow{Prey_j} + Q\cdot XP\cdot\overrightarrow{S} \quad (22)$$

$$\overrightarrow{S_j} = \overrightarrow{Y_L}\cdot\left(\overrightarrow{Y_L}\cdot\overrightarrow{Elite_j} - \overrightarrow{Prey_j}\right),\ j = 1,2,\ldots,n, \quad (23)$$

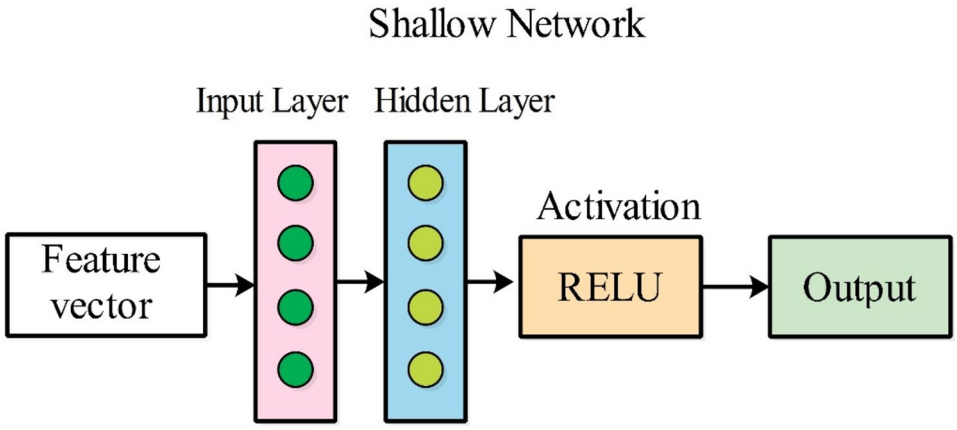

Fig. 8 Shallow wide neural network classifier for GIT classification

Table 3 Classification results of the proposed network-level fusion on the Kvasir V1 dataset

Classifiers	Sensitivity (%)	Precision (%)	FPR	AUC	F1-Score	Accuracy (%)	Time (sec)
NNN	99.47	99.45	0.0	1.00	99.47	99.50	488.53
MNN	99.46	99.45	0.0	1.00	99.45	99.40	464.12
C-SVM	99.41	99.37	0.0	1.00	99.38	99.40	365.35
BNN	99.48	99.46	0.0	1.00	99.46	99.50	386.01
TNN	99.40	99.40	0.0	1.00	99.40	99.40	766.01
SWNN	**99.57**	**99.55**	**0.0**	**1.00**	**99.50**	**99.60**	**292.20**

$$matrices \quad \overrightarrow{Prey_j} = \overrightarrow{Prey_j} + . \quad XP \quad . \quad \overrightarrow{S_{j,}} \quad (24)$$

Where $Q = 0.5$ but can be turned, $XP = \left(1- \quad \frac{t}{T} \right) 2^{\frac{t}{T}}$ holds random values.

Entropy-based sorting

In this phase, the extracted features are processed through an entropy function, which evaluates and sorts them based on their scores. Features with the highest scores are selected for the final feature set. The Marine Predators Algorithm (MPA) operates until a predefined stopping criterion is met; in this work, the maximum number of iterations was set to 200. The MPA algorithm was employed to identify and retain the most relevant features, optimizing the feature selection process. After applying MPA, the selected feature dimensions were reduced to $N \times 1247$ and $N \times 894$, ensuring a more compact and effective representation for subsequent classification tasks.

Shallow neural network classifiers

Finally, the selected features are fed into a Shallow Wide Neural Network (SWNN) classifier. The architecture of the SWNN classifier is illustrated in Fig. 8. The SWNN operates using a forward propagation mechanism, where the input feature vector is passed to the network's input layer. The features are then processed through a single hidden layer, which applies transformations to capture key patterns. The transformed data is subsequently passed through an activation layer to generate the final classification output.

Results and discussion

This section presents the tabular and graphical results of the experiments conducted to evaluate the proposed framework. Two datasets were utilized for the evaluation, as detailed in Sect. "Results and Discussion". Each dataset was divided into training and testing sets using a 50:50 split. Various performance metrics, including F1-score [38], accuracy, sensitivity, precision, computational time, and testing time, were computed to assess the framework's effectiveness. Multiple classifiers were employed for classification, including the Shallow Wide Neural Network (SWNN), Simple Neural Network (NN), and Cubic SVM. All experiments were conducted using MATLAB R2023b on a system equipped with 128 GB of RAM and a 20 GB graphics card Tesla V100.

Dataset Kvasir-VI

Results of network level fusion network

The classification results of the network-level fusion using the Kvasir-V1 dataset are presented in Table 3. The SC-DSAN and CNN-GRU architectures were fused at the network level using depth-wise concatenation. The modified model was trained, and deep features were extracted from the self-attention layer. These features were subsequently fed into multiple classifiers for evaluation. Among all the classifiers, the Shallow Wide

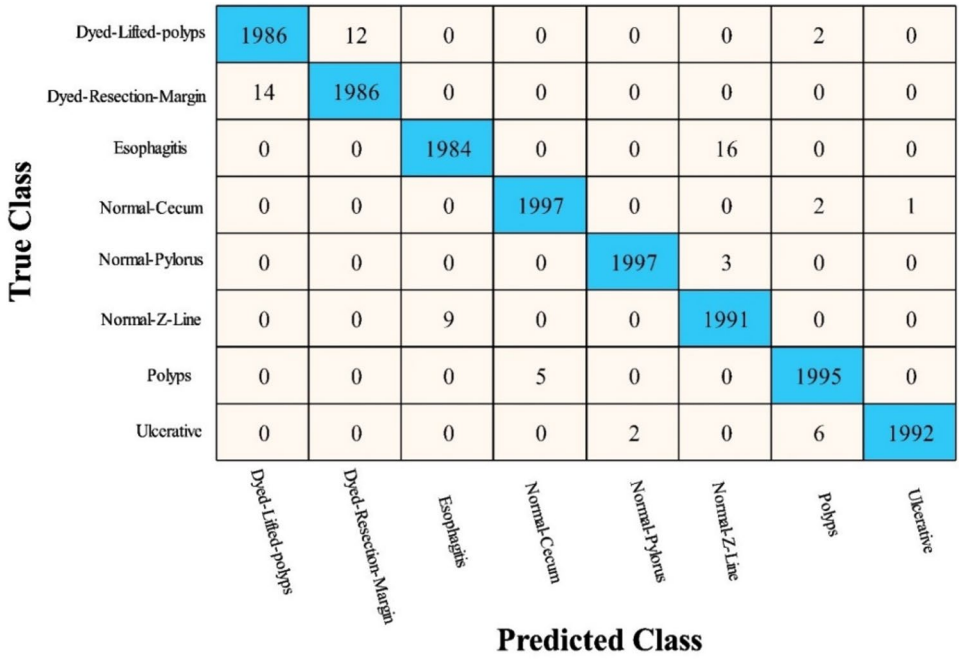

Fig. 9 Confusion matrix of SWNN classifier using Kvasir database V-I

Table 4 Classification results after employing Entrop-controlled MPA algorithm for KvasirV1 dataset

Classifiers	Sensitivity (%)	Precision (%)	FPR	AUC	F1-Score	Accuracy (%)	Time (sec)
NNN	99.46	99.46	0.0	1.00	99.46	99.50	246.03
MNN	99.52	99.52	0.0	1.00	99.52ss	99.50	217.81
CSVM	99.55	99.52	0.0	1.00	99.53	99.50	315.1
BNN	99.52	99.52	0.0	1.00	99.52	99.50	277.65
TNN	99.47	99.47	0.0	1.00	99.47	99.50	512.96
SWNN	**99.66**	**99.62**	**0.0**	**1.00**	**99.63**	**99.60**	**204.06**

Neural Network (SWNN) achieved the highest accuracy, recorded at 99.60%. Additional performance metrics included sensitivity, precision, F1-score, AUC, and FPR, with values of 99.57%, 99.55%, 99.5%, 1, and 0.0, respectively. The same analysis was performed for the other classifiers, with results indicating the superiority of the SWNN. To further validate the performance, a confusion matrix is illustrated in Fig. 9. Computational time was also recorded for each classifier; the SWNN demonstrated the lowest computation time at 292.20 s, whereas the tri-layered neural network exhibited the highest computation time at 766.01 s.

In the next experiment, the best features were selected using an entropy-controlled Marine Predators Algorithm (MPA) optimization and then used for classification. The classification results after applying the entropy-controlled MPA optimization to the Kvasir-V1 dataset are presented in Table 4. From this table, it is evident that the Shallow Wide Neural Network (SWNN) classifier achieved the highest accuracy of 99.60%, with a precision rate of 99.62%, recall rate of 99.66%, F1-score of 99.63%, AUC of 1, and FPR of 0.0. These results are further confirmed by the confusion matrix shown in Fig. 10. For the other classifiers, the same performance metrics were computed. When comparing the results with those in Table 3, it is noted that although there was minimal improvement in accuracy, there was a significant reduction in the testing classification time for all classifiers.

Results of Kvasir VII

The classification results of the proposed network-level model on the Kvasir-VII dataset are presented in Table 5. This table shows that after training on the Kvasir-VII dataset, the prominent features were extracted and passed through multiple classifiers. Among them, the Shallow Wide Neural Network (SWNN) achieved the highest accuracy of 95.13%. Other performance metrics, including TPR, PPV, FPR, AUC, and F1-score, were 95.13%, 95.15%, 0.012, 0.99, and 95.13%, respectively. These metrics were also computed for the other classifiers listed in the table. Figure 11 presents the confusion matrix for the SWNN classifier, which helps validate the performance of the experiment. Additionally, the testing times for all classifiers were recorded, with the SWNN

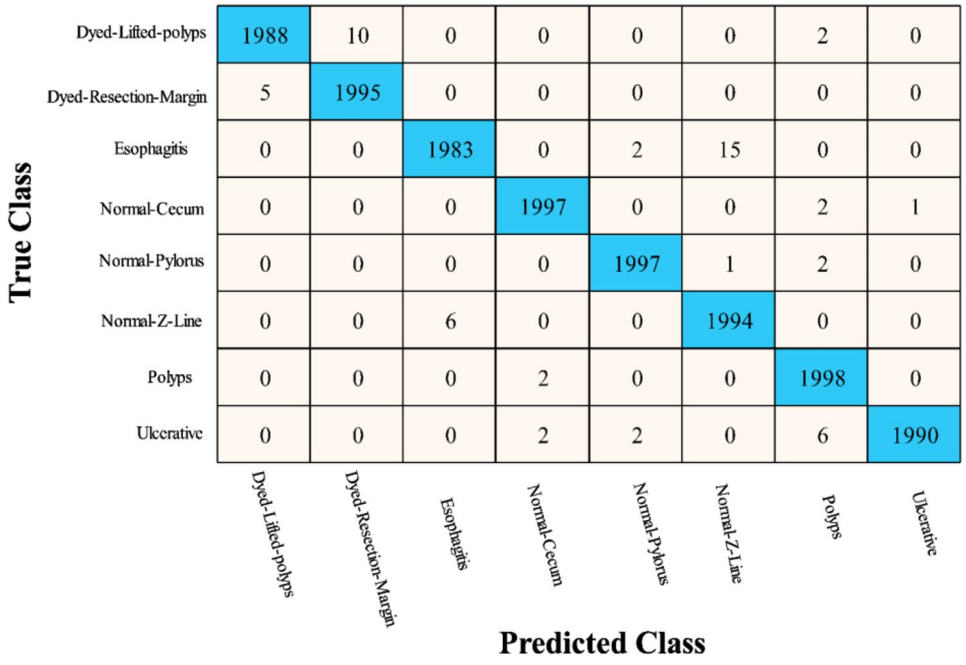

Fig. 10 Confusion matrix of SWNN for the validation of feature selection results on Kvasir V1 dataset

Table 5 Classification results of the proposed network-level fusion on the KvasirVII dataset

Classifiers	TPR	PPV	FPR	AUC	F1-SCORE	Accuracy (%)	Time (sec)
NNN	92.32	92.31	0.01	0.98	92.31	92.30	477.3
MNN	92.92	92.96	0.01	0.98	92.93	93.00	341.1
CSVM	92.42	92.42	0.01	0.98	92.42	92.40	205.8
BNN	91.43	91.42	0.007	0.97	91.42	91.40	261.3
TNN	91.23	91.26	0.012	0.98	91.24	91.20	334.6
SWNN	**95.13**	**95.15**	**0.012**	**0.99**	**95.13**	**95.10**	**149.69**

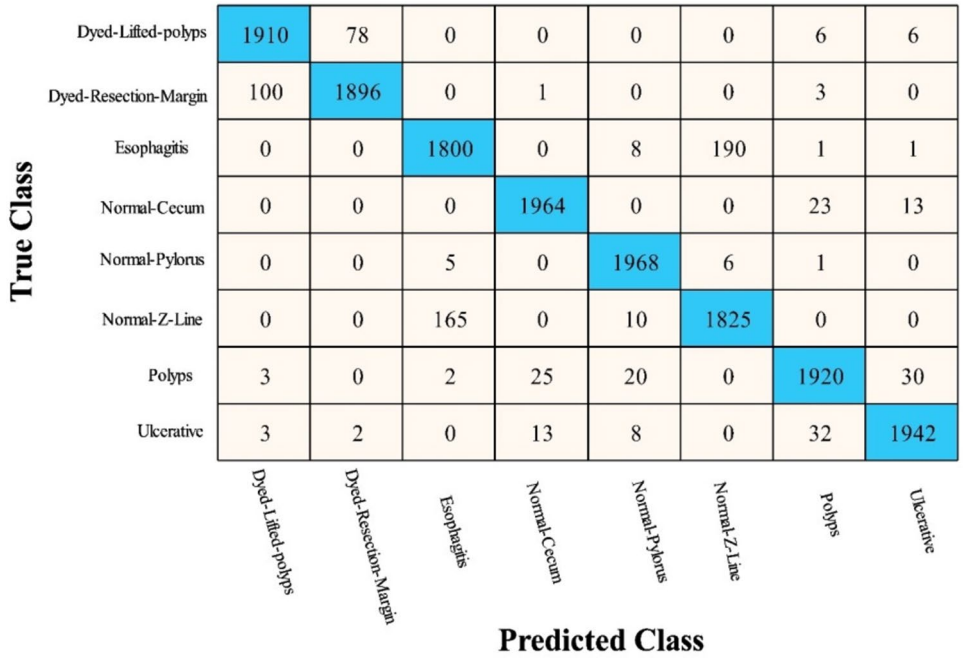

Fig. 11 SWNN classifier confusion matrix for network level fusion model on Kvasir VII dataset

Table 6 Classification results of the proposed architecture after employing optimization algorithm on the KvasirVII dataset

Classifiers	TPR	PPV	FPR	AUC	F1-SCORE	Accuracy (%)	Time (sec)
NNN	96.01	95.98	0.004	0.98	95.99	96.00	119.8
MNN	96.68	96.67	0.003	0.98	96.67	96.70	137.1
CSVM	96.63	96.63	0.005	0.99	96.63	96.60	138.6
BNN	95.83	95.46	0.004	0.98	95.64	95.80	128.3
TNN	94.95	94.20	0.005	0.98	94.57	94.90	103.7
SWNN	**96.6**	**96.61**	**0.004**	**1.00**	**96.60**	**96.60**	**94.34**

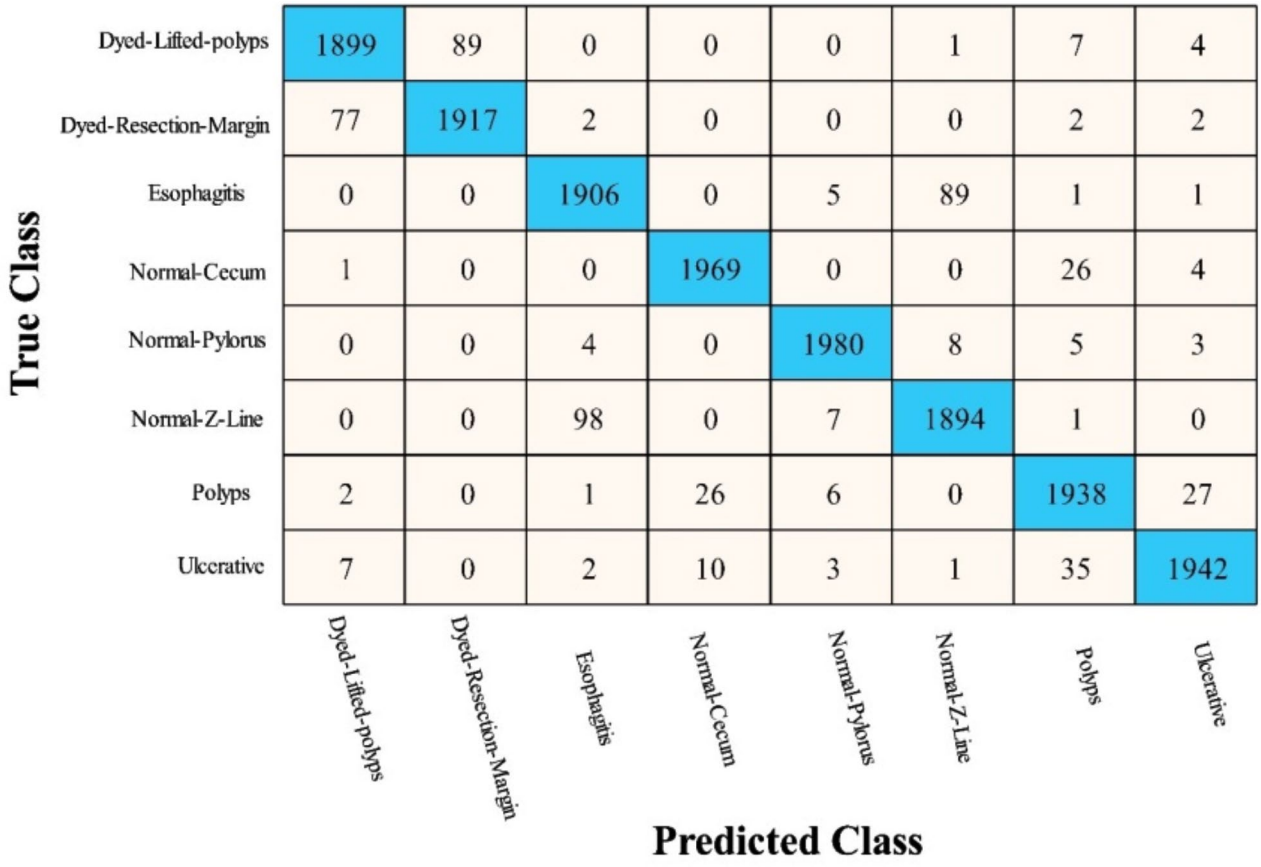

Fig. 12 SWNN confusion matrix for MPA algorithm on Kvasir VII dataset

classifier achieving the shortest testing time of 149.69 s, while the Narrow Neural Network (NNN) classifier took the longest at 477.3 s.

In the next experiment, the best features are selected to enhance precision and accuracy while reducing classification time. To achieve this, the features extracted from the network-level fusion are passed through the Entropy-controlled Marine Predators Algorithm (MPA) for optimal feature selection. The selected features are then used in the classification phase, with results shown in Table 6. According to the table, the Shallow Wide Neural Network (SWNN) achieved the highest accuracy of 96.60%, with a TPR of 96.6%, PPV of 96.61%, F1-score of 96.60%, FPR of 0.004, and AUC of 1.00. These performance metrics were also computed for the other classifiers listed in the table.

To further validate the performance of the SWNN classifier, the confusion matrix is presented in Fig. 12. This figure highlights a significant reduction in the false negative rate. Additionally, the computational time for each classifier was recorded, with the SWNN classifier showing the shortest execution time. Overall, the optimization technique not only improved classification accuracy but also reduced testing time, demonstrating its effectiveness.

Discussion

This section provides a detailed discussion of the proposed fused architecture and optimization technique. The performance of the proposed architecture is evaluated based on the following points: (i) comparison of the fused architecture's performance with existing pre-trained models; (ii) comparison of the optimization algorithm results with state-of-the-art (SOTA) techniques; (iii) statistical analysis using the standard error of the mean for the selected datasets; and, finally, a comparison of the proposed method's results with SOTA techniques.

Figure 2 illustrates the proposed architecture, which integrates data augmentation, network-level model fusion, hyperparameter selection via Bayesian Optimization, and optimal feature selection. The classification results for the proposed network-level fusion are presented in Tables 3 and 5, while the optimization results are discussed in Tables 4 and 6. To further validate the results, confusion matrices are provided for the best-performing models, as shown in Figs. 8, 9, 10 and 11.

The proposed framework shows high scalability through efficient network-level fusion, integrating Sparse Convolution operation, and CNN-GRU, maintaining high precision. The procedure of Bayesian optimization to optimize hyperparameters and the entropy-controlled marine predator algorithm to select best features [39] improved the framework adaptation to large-scale. In addition, the proposed framework has the ability to obtain the high performance and integration with GradCam interpretation [40, 41], make it promising for the clinical applications. However, computational requirements remain a concern for deployment in resource-limited environments, which requires quantification to ensure a wider adoption in healthcare.

Ablation study

A detailed ablation study is conducted to analyze the performance of the proposed fused model. In this study, several pre-trained models are employed and evaluated on the selected datasets. Table 7 presents the numerical results of the ablation study, including accuracy and the total number of learnable parameters (in millions).

Table 7 Comparison of the proposed network-level fusion architecture with SOTA pre-trained models based on the accuracy of the selected datasets of this work

Network	Kvasir V1 (%)	Kvasir VII (%)	Parameters (M)	Parameter memory (PM)
Proposed	**99.60**	**95.10**	**14.7**	**51 MB**
Inception V3	96.40	92.86	23.9	91 MB
DenseNet201	95.10	93.04	20.0	77 MB
Resnet18	94.50	91.28	11.7	45 MB
Resnet50	94.86	90.20	25.5	98 MB
Resnet101	92.36	90.14	44.6	171 MB
Xception	94.90	93.68	22.9	88 MB
NasnetLarge	95.60	93.60	88.9	340 MB

In this table, the Inception V3 model achieved accuracies of 96.40% and 92.86% on the Kvasir V1 and Kvasir VII datasets, respectively, with 23.9 million learnable parameters. The DenseNet201 model achieved accuracies of 95.10% and 93.04%, with 20.0 million learnable parameters. The pre-trained models ResNet18, ResNet50, and ResNet101 achieved accuracies of 94.50% and 91.28%, 94.86% and 90.20%, and 92.36% and 90.14% for Kvasir V1 and Kvasir VII, respectively, with learnable parameters of 11.7 million, 25.5 million, and 44.6 million, respectively. Additionally, the NasNetLarge model achieved accuracies of 95.60% and 93.60%, with 88.9 million learnable parameters. The results show that the proposed fused architecture outperforms these state-of-the-art models in terms of accuracy while requiring fewer learnable parameters.

The proposed network has a significant advantage in parameter memory efficiency compared to some state-of-the-art models. It only requires 51 MB of parameter memory, and hits models such as Inception V3 (91 MB), DenseNet201 (77 MB), and Xception (88 MB). This efficiency is even more notable compared to ResNet50 (98 MB) and NasNetLarge (340 MB), which is very parameter-intensive. Although memory is low, the proposed architecture achieves superior classification accuracy and indicates the best balance between model complexity and performance.

In another experiment, several state-of-the-art (SOTA) optimization algorithms, including WOA, BCO, ACO, AntLion, MPO, Crow Search, and Grey Wolf, were employed for feature selection. The numerical results of this experiment are presented in Fig. 13. The accuracy obtained by these algorithms on the Kvasir VI dataset was 96.24%, 94.2%, 95.12%, 95.06%, 96.88%, 95.86%, and 94.29%, respectively. For the Kvasir VII dataset, the accuracies were 92.02%, 91.63%, 92.7%, 93.5%, 94.1%, 92.01%, and 93.42%, respectively. In comparison, the proposed algorithm achieved an accuracy of 99.6% on the Kvasir VI dataset and 96.6% on the Kvasir VII dataset. When comparing the accuracy of the SOTA methods to that of the proposed approach, it is clear that the proposed method delivers superior performance. Figure 14 displays the prediction results from the proposed model. The prediction image was generated by the proposed fused model, and the region extraction was performed using the heatmap generated by the GradCAM approach.

In this ablation study, a comparative analysis was performed between the baseline DenseNet201 model and variants of the proposed models, as presented in Table 8. The baseline model, which has 20 million parameters, was trained for 11 h and 6 min on the Kvasir-VI dataset.

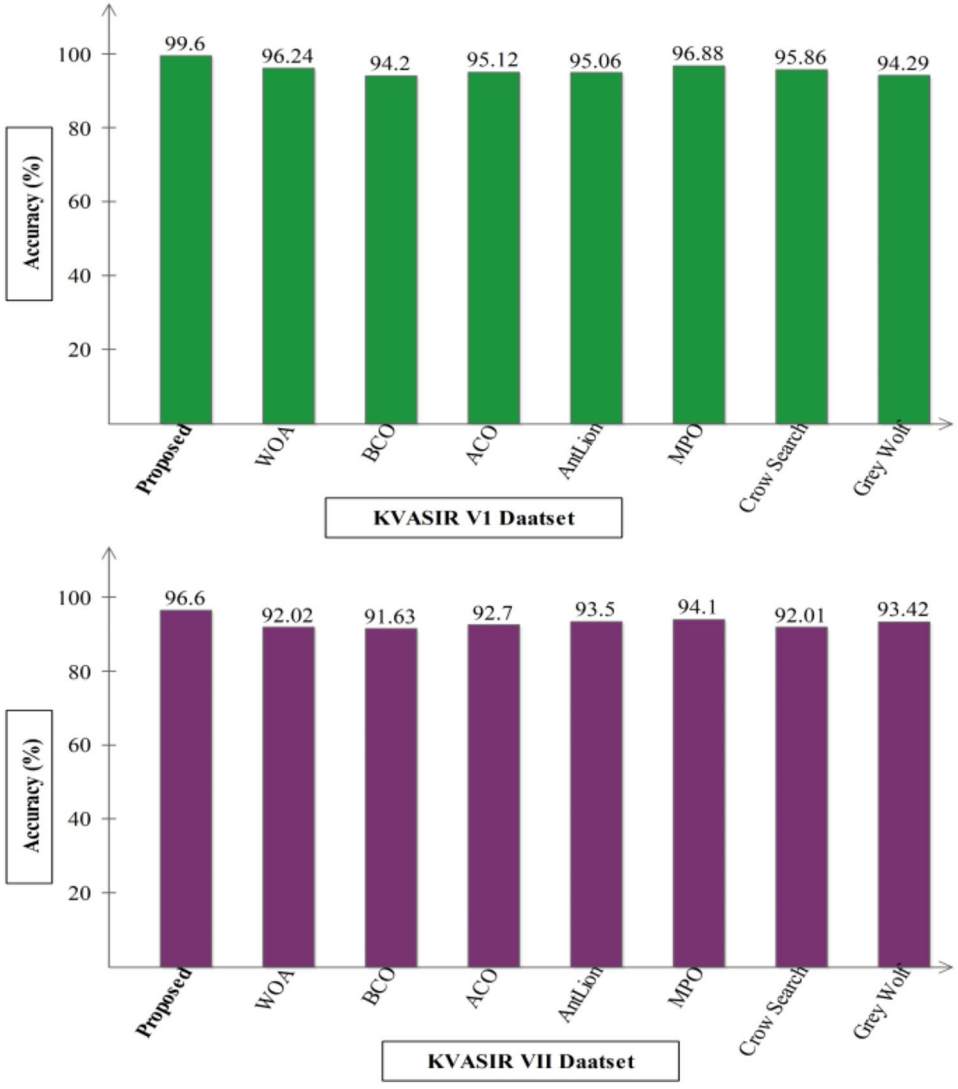

Fig. 13 Comparison of improved optimization algorithm with SOTA techniques

Training on the Kvasir-VII dataset took 9 h and 18 min due to the traditional convolutional operations.

Next, both datasets were trained using the DenseNet201 model with the self-attention layer, which required 12 h and 15 min for training, with 20.6 million parameters. The DenseNet201 model was then modified by replacing the convolutional layers with sparse convolution layers. This proposed model trained for 9 h and 5 min on the Kvasir-VI dataset and 8 h and 16 min on the Kvasir-VII dataset, with 12.4 million parameters.

Finally, network-level fusion was applied in the last variant. This model, with 14.7 million parameters, required 10 h and 21 min to train on the Kvasir-VI dataset and 9 h and 48 min on the Kvasir-VII dataset. The analysis reveals that sparse convolutional operations significantly reduce the complexity and number of parameters in the proposed network, which directly impacts training time and makes the network less computationally expensive.

In this next phase, the Five-fold cross-validation using network-level fusion results is described in Table 9. According to the table, the proposed network was consistently improved with an average training accuracy of 98.5%, precision of 97.3%, and recall of 96.42% on Kvasir V1. Each fold maintained a substantial accuracy of more than 97.5%, reaching the highest level of 99.4% in 5 fold. At the same time, Kvasir V2 produced relatively lower but stable results, with an average training accuracy of 94.6%, a precision of 93.6%, and a recall of 93.9%. The highest accuracy of Kvasir V2 was 95.3% in fold 5, while the lowest was 93.6% in fold 3. These results suggest that the proposed network-level fusion approach is more effective than the proposed Kvasir V1 and demonstrates its sensitivity to specific datasets and superior classification capabilities.

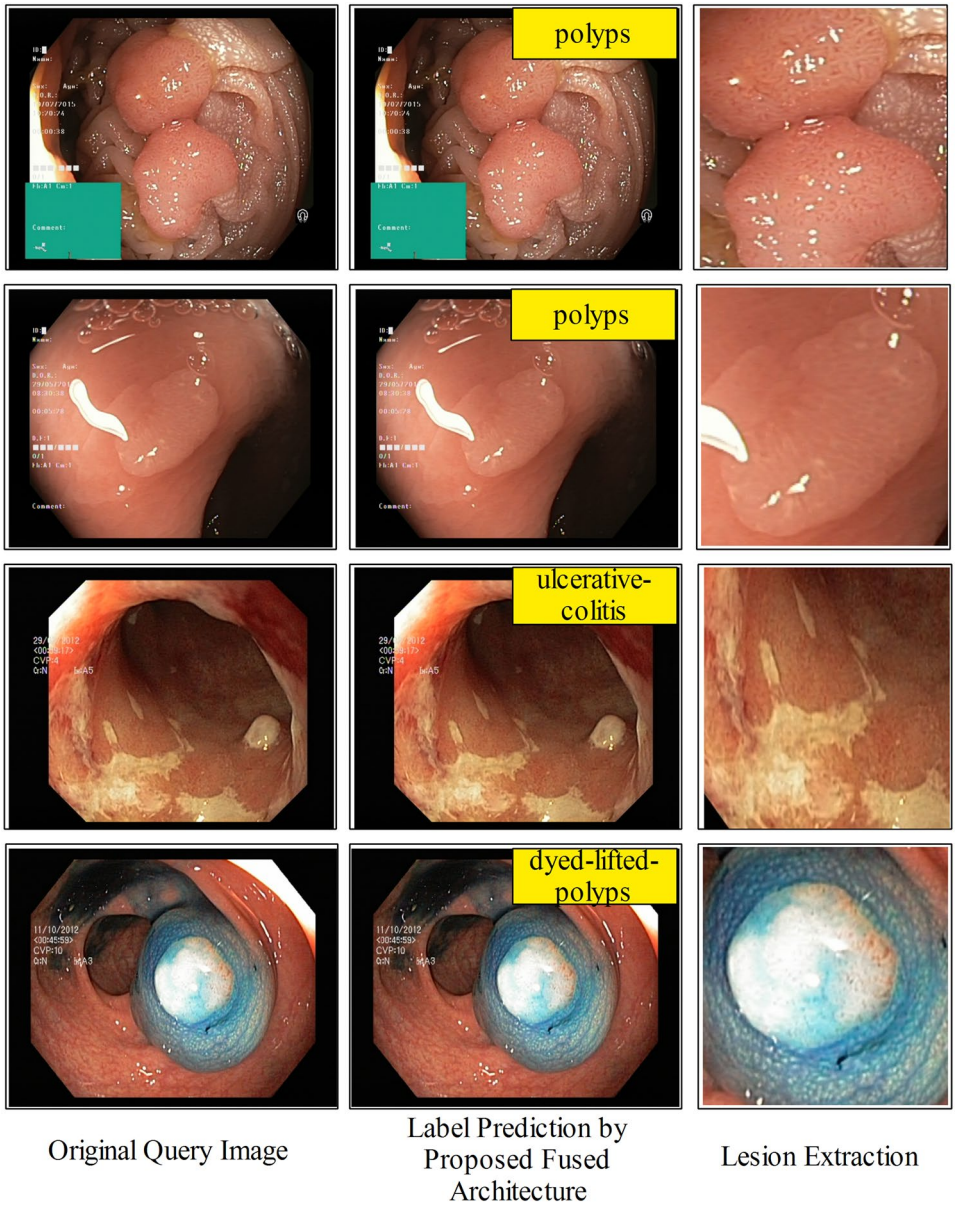

Fig. 14 Proposed architecture labelled prediction and lesion region extraction

Comparison with SOTA

In this section, a comprehensive comparison with state-of-the-art (SOTA) frameworks is presented in Table 10. Gamage et al. [26] proposed an ensemble deep learning framework for classifying the Kvasir-VII dataset, achieving the highest accuracy of 90.74%. In 2021, Mubarak et al. [27] and Hmoud Al-Adhaileh et al. [28] introduced deep learning-based frameworks using transfer learning with pre-trained CNNs. Both frameworks utilized the Kvasir-VI dataset and achieved accuracies of 94.46% and 97.0%, respectively. In 2022, Ahmed et al. [29] presented a CNN-based denoising model trained on the pre-trained AlexNet using the Kvasir-VI dataset, achieving an accuracy of 90.17%. Khan et al. [30] developed an automatic deep learning framework combined with hybrid crow-moth optimization for the classification and identification of stomach diseases, achieving the highest accuracy of 97.85%. In 2024, Farooq et al. [31] presented deep learning models such as DarkNet52 and Xception, coupled with dragonfly optimization for feature selection, for classifying gastrointestinal tract syndromes using the Kvasir-VI dataset. Their framework achieved an accuracy of 98.25%. Our proposed framework outperforms these SOTA techniques, achieving 99.60% accuracy on the Kvasir-VI dataset and 95.10% on the Kvasir-VII dataset, demonstrating a significant improvement in accuracy.

Table 8 Analysis of proposed model based on various variants

Kvasir-VI			
Variants	**Parameters**	**V(Accuracy)**	**Training Time**
Baseline DensNet201	20 M	94.92	11 h 6 m
DensNet201 + SA	20.6 M	95.14	12 h 15 m
Proposed SC-DSAN	12.4 M	97.82	9 h 40 m
Proposed Network Level Fusion	14.7 M	99.19	10 h 21 m
Kvasir-VII			
Base DensNet201	-	90.94	9 h 18 m
DensNet201 + SA	-	92.66	11 h 27 m
Proposed SC-DSAN	-	94.18	8 h 16 m
Proposed Network Level Fusion	-	95.24	9 h 48 m

Table 9 Analysis of proposed model based on various folds

Proposed Network Level Fusion					
Fold	**Kvasir V1**	**Kvasir V2**	**Tr(Accuracy)**	**Precision**	**Recall**
1	✓	-	98.6	97.2	96.4
	-	✓	95.1	94.6	93.8
2	✓	-	98.4	97.4	96.6
	-	✓	94.8	93.3	93.1
3	✓	-	97.5	96.1	95.4
	-	✓	93.6	92.1	91.2
4	✓	-	98.9	97.3	96.1
	-	✓	94.5	93.4	93.0
5	✓	-	99.4	98.5	97.6
	-	✓	95.3	94.6	93.9
Mean	✓	-	98.5	97.3	96.42
	-	✓	94.6	93.6	93.0

Table 10 Comparison of proposed technique with SOTA methods on selected datasets

Authers	**Year**	**Dataset**	**Accuracy (%)**
C. Gamage et al. [42]	2019	Kvasir-V2	90.74
D. Mubarak et al. [43]	2021	Kvasir-V1	94.46
M. Adhaileh et al. [44]	2021	Kvasir-V1	97.00
A. Ahmed et al. [45]	2022	Kvasir-V1	90.17
M. A. Khan et al. [46]	2022	Kvasir-V1	97.85
Khan Farooq et al. [47]	2024	Kvasir-VI	98.25
Proposed		Kvasir-V1	99.60
Proposed		Kvasir-VII	95.10

Conclusion

In this study, we proposed a fully automated deep-learning framework for diagnosing and classifying gastrointestinal diseases from wireless-capsule endoscopic images. The framework integrates a CNN-GRU model with a self-attention-enhanced SC-DSAN architecture, complemented by dataset augmentation and optimized feature extraction using Entropy MPO (EMPO). Our experimental results demonstrate the framework's high accuracy, achieving 99.60% on the Kvasir-V1 dataset and 95.10% on the Kvasir-V2 dataset. Comparisons with recent studies and pre-trained networks highlight our framework's superior performance in terms of accuracy and precision. Additionally, using GradCAM visualization methods improves the model's interpretability, providing valuable insights into its decision-making process.

Despite these promising results, this study has limitations. The framework was evaluated solely on the Kvasir-V1 and Kvasir-V2 datasets, which may not encompass the full spectrum of gastrointestinal diseases. Additionally, the CNN-GRU model's computational complexity may pose challenges for its deployment in resource-constrained environments.

Future work will focus on addressing these limitations. We plan to extend the evaluation to a broader range of datasets to ensure the model's validity across diverse clinical scenarios. We will explore model optimization techniques such as pruning and quantization to reduce computational complexity, which could improve deployment feasibility and generalization. Finally, we aim to develop advanced interpretability methods that offer deeper insights into the model's decisions, enhancing its acceptance in clinical settings.

Acknowledgements

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(*MSIT) (No.2018R1A5A7059549). This work was supported through Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2025R508), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. The authors extend their appreciation to the Deanship of Research and Graduate Studies at King Khalid University for funding this work through Large Research Project under grant number RGP.2/379/45.

Author contributions

All authors contributed equally in this work. All authors agreed to submit work in this reputed journal.

Funding

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(*MSIT) (No.2018R1A5A7059549). This work was supported through Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2025R508), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. The authors extend their appreciation to the Deanship of Research and Graduate Studies at King Khalid University for funding this work through Large Research Project under grant number RGP.2/379/45.

Data availability

The datasets of this work such as Kvasir V1 and V2 are publically available for the research purposes on the following link (https://datasets.simula.no/kvasir/). All researchers that will use this dataset, must cite this paper: K. Pogorelov, K. R. Randel, C. Griwodz, S. L. Eskeland, T. de Lange, D. Johansen, et al., "Kvasir: A multi-class image dataset for computer aided gastrointestinal disease detection," in *Proceedings of the 8th ACM on Multimedia Systems Conference*, 2017, pp. 164–169.

Declarations

Ethical approval

Not applicable.

Consent to participate
Not applicable.

Competing interests
The authors declare no competing interests.

Author details
[1]Department of Computer Science and Engineering, College of Computer Engineering and Science, Prince Mohammad Bin Fahd University, Al-Khobar, KSA, Kingdom of Saudi Arabia
[2]Department of Computer Science, HITEC University, Taxila, Pakistan
[3]Centre of Real Time Computer Systems, Kaunas University of Technology (KTU), Kaunas, Lithuania
[4]Department of Computer Engineering, College of Computer Science, King Khalid University, Abha 61413, Saudi Arabia
[5]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O.Box 84428, Riyadh 11671, Saudi Arabia
[6]HYU Center for Computational Social Science, Hanyang University, Seoul, South Korea
[7]Department of Computer Science, Hanynag University, seoul 01000, Korea, Republic of (South Korea)

Received: 13 December 2024 / Accepted: 10 March 2025
Published online: 31 March 2025

References

1. Bajhaiya D, Unni SN. Deep learning-enabled detection and localization of Gastrointestinal diseases using wireless-capsule endoscopic images. Biomed Signal Process Control. 2024;93:106125.
2. Hussain S, Mubeen I, Ullah N, Shah SSUD, Khan BA, Zahoor M, et al. Modern diagnostic imaging technique applications and risk factors in the medical field: a review. Biomed Res Int. 2022;2022:5164970.
3. Panayides AS, Amini A, Filipovic ND, Sharma A, Tsaftaris SA, Young A, et al. Ai in medical imaging informatics: current challenges and future directions. IEEE J Biomedical Health Inf. 2020;24:1837–57.
4. Peery AF, Crockett SD, Murphy CC, Lund JL, Dellon ES, Williams JL, et al. Burden and cost of gastrointestinal, liver, and pancreatic diseases in the United States: update 2018. Gastroenterology. 2019;156:254-272.e11.
5. Machicado JD, Greer JB, Yadav D. Epidemiology of gastrointestinal diseases. Geriatric Gastroenterol. 2020;11:1–21.
6. Bandl L, Billroth T, Börner E, Breisky A, Charpentier ALA, Chrobak R, et al. Cyclopædia of obstetrics and gynecology, vol. 10: W. Wood & Company, 1887.
7. Galimzhanov A, Matetic A, Tenekecioglu E, Mamas MA. Prediction of clinical outcomes after percutaneous coronary intervention: Machine-learning analysis of the National inpatient sample. Int J Cardiol. 2023;392:131339.
8. Bray F, Ferlay J, Soerjomataram I, Siegel RL, Torre LA, Jemal A. Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. Cancer J Clin. 2018;68:394–424.
9. Wilkinson C. Older Australians: trends and impacts of alcohol and other drug use. WA: National Drug Research Institute, Curtin University Perth; 2018.
10. El-Ghany SA, Mahmood MA, Abd El-Aziz A. An accurate deep Learning-Based Computer-Aided diagnosis system for Gastrointestinal disease detection using wireless capsule endoscopy image analysis. Appl Sci. 2024;14:10243.
11. Jothiraj S, Kandaswami JA. Localization and semantic segmentation of polyp in an effort of early diagnosis of colorectal cancer from wireless capsule endoscopy images, in 2022 Seventh International Conference on Parallel, Distributed and Grid Computing (PDGC), 2022, pp. 749–754.
12. Mehedi IM, Rao KP, Alotaibi FM, Alkanfery HM. Intelligent wireless capsule endoscopy for the diagnosis of gastrointestinal diseases, Diagnostics. 2023;13:1445.
13. Cao Q, Deng R, Pan Y, Liu R, Chen Y, Gong G, et al. Robotic wireless capsule endoscopy: recent advances and upcoming technologies. Nat Commun. 2024;15:4597.
14. Raut V, Gunjan R, Shete VV, Eknath UD. Gastrointestinal tract disease segmentation and classification in wireless capsule endoscopy using intelligent deep learning model. Comput Methods Biomech Biomedical Engineering: Imaging Visualization. 2023;11:606–22.
15. Obayya M, Al-Wesabi FN, Maashi M, Mohamed A, Hamza MA, Drar S, et al. Modified salp swarm algorithm with deep learning based Gastrointestinal tract disease classification on endoscopic images. IEEE Access. 2023;11:25959–67.
16. Hossain MM, Mary MM, Islam MR. Optimizing Skin Lesion Segmentation with UNet and Attention-Guidance Utilizing Test Time Augmentation. In: 2024 6th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT), 2024, pp. 568–573.
17. Sharma R, Lamba C. Advancing Gastrointestinal disease detection through artificial intelligence: A comprehensive analysis. Int J Intell Syst Appl Eng. 2024;12:514–8.
18. Ali H, Muzammil MA, Dahiya DS, Ali F, Yasin S, Hanif W, et al. Artificial intelligence in Gastrointestinal endoscopy: a comprehensive review. Annals Gastroenterol. 2024;37:133.
19. Hossain MM, Nahiduzzaman M, Islam MR, Islam MR, Ahsan M, et al. A review on brain tumor segmentation based on deep learning methods with federated learning techniques. Comput Med Imaging Graph. 2023;3:102313.
20. Malik H, Naeem A, Sadeghi-Niaraki A, Naqvi RA, Lee S-W. Multi-classification deep learning models for detection of ulcerative colitis, polyps, and dyed-lifted polyps using wireless capsule endoscopy images. Complex Intell Syst. 2024;10:2477–97.
21. Ma L, Su X, Ma L, Gao X, Sun M. Deep learning for classification and localization of early gastric cancer in endoscopic images. Biomed Signal Process Control. 2023;79:104200.
22. Nouman Noor M, Nazir M, Khan SA, Song O-Y, Ashraf I. Efficient gastrointestinal disease classification using pretrained deep convolutional neural network,. Electronics. 2023;12:1557.
23. Ghaleb Al-Mekhlafi Z, Mohammed Senan E, Alshudukhi JS, Abdulkarem Mohammed B. Hybrid techniques for diagnosing endoscopy images for early detection of gastrointestinal disease based on fusion features. Int J Intell Syst. 2023;2023:8616939
24. Bhardwaj P, Kumar S, Kumar Y. A comprehensive analysis of deep learning-based approaches for the prediction of Gastrointestinal diseases using multi-class endoscopy images. Arch Comput Methods Eng. 2023;30:4499–516.
25. Demirbaş AA, Üzen H, Fırat H. Spatial-attention convmixer architecture for classification and detection of Gastrointestinal diseases using the Kvasir dataset. Health Inform Sci Syst. 2024;12:32.
26. Li X, Wu Q, Wu K. Wireless capsule endoscopy anomaly classification via dynamic multi-task learning. Biomed Signal Process Control. 2025;100:107081.
27. Nayyar Z, Khan MA, Alhussein M, Nazir M, Aurangzeb K, Nam Y, et al. Gastric tract disease recognition using optimized deep learning features. Comput Mater Contin. 2021;68:2041–56.
28. Vania M, Tama BA, Maulahela H, Lim S. Recent advances in applying machine learning and deep learning to detect upper Gastrointestinal tract lesions. IEEE Access. 2023;8:1–18.
29. Faramarzi A, Heidarinejad M, Mirjalili S, Gandomi AH. Marine predators algorithm: A nature-inspired metaheuristic. Expert Syst Appl. 2020;152:113377.
30. Selvaraj J, Jayanthy A. Design and development of artificial intelligence-based application programming interface for early detection and diagnosis of colorectal cancer from wireless capsule endoscopy images. Int J Imaging Syst Technol. 2024;34:e23034.
31. Al-Otaibi S, Rehman A, Mujahid M, Alotaibi S, Saba T. Efficient-gastro: optimized EfficientNet model for the detection of gastrointestinal disorders using transfer learning and wireless capsule endoscopy images, *PeerJ Computer Science*, vol. 10, p. e1902, 2024.
32. Kalinin AA, Iglovikov VI, Rakhlin A, Shvets AA. Medical image segmentation using deep neural networks with pre-trained encoders. Deep Learn Appl. 2020;21:39–52.
33. Pogorelov K, Randel KR, Griwodz C, Eskeland SL, de Lange T, Johansen D et al.,, Kvasir: A multi-class image dataset for computer aided gastrointestinal disease detection, in Proceedings of the 8th ACM on Multimedia Systems Conference, 2017, pp. 164–169.
34. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4700–4708.
35. Zhu Y, Newsam S. Densenet for dense flow, in 2017 IEEE international conference on image processing (ICIP), 2017, pp. 790–794.
36. Ravanelli M, Brakel P, Omologo M, Bengio Y. Light gated recurrent units for speech recognition. IEEE Trans Emerg Top Comput Intell. 2018;2:92–102.
37. Ramadevi P, Das R. An extensive analysis of machine learning techniques with hyper-parameter tuning by bayesian optimized SVM kernel for the detection of human lung disease. IEEE Access. 2024;6:1–14.

38. Selvaraj J, Jayanthy A. Automatic polyp semantic segmentation using wireless capsule endoscopy images with various convolutional neural network and optimization techniques: A comparison and performance evaluation. Biomedical Engineering: Appl Basis Commun. 2023;35:2350026.
39. Shafi FB, Nahiduzzaman M, Ayari MA, Khandakar A. Interpretable deep learning architecture for Gastrointestinal disease detection: A Tri-stage approach with PCA and XAI. Comput Biol Med. 2025;185:109503.
40. Nahiduzzaman M, Islam MR, Naznine M, Ayari MA, Khandakar A, et al. Detection of various Gastrointestinal tract diseases through a deep learning method with ensemble ELM and explainable AI. Expert Syst Appl. 2024;256:124908.
41. Islam MR, Nahiduzzaman M, Karim MJ, Ayari MA, Khandakar A. Automated detection of colorectal polyp utilizing deep learning methods with explainable AI. IEEE Access. 2024;26:1–18.
42. Gamage C, Wijesinghe I, Chitraranjan C, Perera I. GI-Net: anomalies classification in gastrointestinal tract through endoscopic imagery with deep learning, in 2019 Moratuwa Engineering Research Conference (MERCon), 2019, pp. 66–71.
43. Mubarak D. Classification of early stages of esophageal cancer using transfer learning. Irbm. 2022;43:251–8.
44. Hmoud Al-Adhaileh M, Mohammed Senan E, Alsaade W, Aldhyani THH, Alsharif N, Abdullah Alqarni A et al.,., Deep learning algorithms for detection and classification of gastrointestinal diseases, Complexity, vol. 2021, pp. 1–12, 2021.
45. Ahmed A. Classification of gastrointestinal images based on transfer learning and denoising convolutional neural networks, in Proceedings of International Conference on Data Science and Applications: ICDSA 2021, Volume 1, 2022, pp. 631–639.
46. Khan MA, Muhammad K, Wang S-H, Alsubai S, Binbusayyis A, Alqahtani A, et al. Gastrointestinal diseases recognition: a framework of deep neural network and improved moth-crow optimization with Dcca fusion. Hum -Cent Comput Inf Sci. 2022;12:25.
47. Khan ZF, Ramzan M, Raza M, Khan MA, Iqbal K, Kim T et al. Deep convolutional neural networks for accurate classification of Gastrointestinal tract syndromes. Computers Mater Continua, 78, 2024.

Publisher's note