SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V12I8P128

Volume 12 Issue 8, 316-333, August 2025

Original Article

CT-CXR-Net: Optimal Deep Learning Framework for Dual-Modal COVID-19 Classification

Anitha Patibandla[1], Kirti Rawal[2]

[1,2]*Department of SEEE, Lovely Professional University, Punjab, India.*

[1]*Corresponding Author : anithakamal.ghanta@gmail.com*

Received: 15 June 2025 Revised: 16 July 2025 Accepted: 17 August 2025 Published: 30 August 2025

Abstract *- Since the onset of the COVID-19 pandemic, more than 700 million people have been impacted by the disease, and more than 7 million people have died, underlining the necessity of quickly and efficiently identifying and diagnosing the disease in controlling its spread. Despite the immense progress, conventional methods of testing usually experience the drawback of being too slow, accessible, and imprecise, especially within resource-limited settings. Existing COVID-19 classification models based on Artificial Intelligence (AI) are affected by noise interference of the medical images, sub-optimal segmentation, and inefficient feature selection, all of which contribute to a low reliability of diagnosis. To overcome these challenges, the novel CT-CXR-COVID-19 Classification Network (CT-CXR-Net) method begins with Block-Matching and 3D filtering (BM3D) denoising to effectively eliminate complex noise patterns, ensuring high-quality input for further analysis. An Optimal U-Net (OU-Net) segmentation model is employed, whose loss is minimized using Modified Grey Wolf Optimization (MGWO), leading to precise lung region extraction. Subsequently, ResNet50 is utilized for deep feature extraction, capturing complex and informative patterns from both Computer Tomography (CT) and Chest X Ray (CXR) images. To reduce feature dimensionality and enhance classification performance, Improved Brown Bear Optimization (IBBO) is adopted for optimal feature selection. Finally, a Ridge Classifier provides robust and efficient classification, maintaining a balance between bias and variance. This approach achieves exceptional results on separate datasets, with the CT dataset recording 100% accuracy, while the CXR dataset achieves 99.30% accuracy, 99.60% precision, and 99.95% recall and F1-score, demonstrating its potential for reliable and high-performance COVID-19 diagnosis.*

Keywords *- BM3D denoising, COVID-19 detection, medical image classification, Modified Grey Wolf Optimization, U-Net segmentation.*

1. Introduction

Since late 2019, COVID-19 has surpassed 700 million confirmed cases and produced over 7 million deaths globally, marking one of the most devastating pandemics in modern history [1]. Its rapid transmission, driven by asymptomatic spread and variants of concern, has placed unparalleled pressure on healthcare systems, revealing a critical need for fast, accurate, and scalable diagnostic tools to manage disease progression and prevent further casualties.

Conventional diagnostic approaches [2], such as radiologist-interpreted imaging, have faced significant challenges. RT-PCR testing delays and variable Sensitivity hinder timely detection, particularly in asymptomatic and early-stage cases [3]. Meanwhile, radiological analysis of chest CT and CXR scans depends heavily on clinician expertise, leading to inconsistent results, high workloads, and limited access, especially in under-resourced settings. To address these limitations, several companies have developed AI-integrated medical imaging solutions. Infervision's AI-powered CT tool [4], deployed in over 34 Chinese hospitals, flagged signs of COVID-19 pneumonia within seconds from thousands of scans. US-grown Aidoc's AI triage system, FDA-approved for chest CT scans, is now in use in more than 1,500 imaging centers globally, including Yale New Haven and Cedars-Sinai. Additionally, Alibaba's cloud-based CT analysis system [5] reached diagnosis speeds of around 20 seconds with 96 % accuracy across 26 hospitals. These examples demonstrate industry-scale strides in integrating AI for rapid and accurate imaging support. Hospitals worldwide have begun adopting AI-augmented imaging tools to assist clinical workflows. Zhongnan Hospital of Wuhan University deployed Infervision's CT software to triage and isolate suspected cases early in the outbreak [6]. In South Korea and Brazil, Lunit's CXR AI solution [7] supported radiologists in high-volume COVID-19 screening. Meanwhile, Minnesota hospitals conducted prospective validation of interpretable CXR AI tools across 12 institutions, demonstrating specificity and sensitivity improvements during real-time clinical use. Academic institutions and multinational hospital networks have pursued rigorous AI research [8] in thoracic imaging-a deep-learning CT community pneumonia across diverse patient cohorts. An ensemble CXR network in Italy achieved 98 % sensitivity and 94 % AUC, outperforming radiologists on independent test sets [9]. A Minnesota-based prospective observational study found AI CXR tools supported clinical decision-making across 12 hospitals, though emphasized they should augment, not replace, clinical judgment. These efforts showcase AI's growing role in diagnostic accuracy and workflow integration.

The remaining part of the paper is organized as follows: Section 2 has an outline of the literature survey with a highlight of the available methods. Section 2 clears up the suggested CT-CXR-Net structure, Section 3 examines the outcomes of experiments, and Section 4 ends the study.

2. Literature Survey

This section reviews recent methodologies exhausting both CXR and CT datasets, highlighting their strengths and limitations. It helps identify critical research gaps that guide the novel contributions of the proposed work.

2.1. Related Work on CXR Dataset

Patnaik et al. [10] proposed a Convolutional Neural Network (CNN)-based model named ReSqNet; they included residual and squeeze-excitation blocks to enhance feature extraction and spatial attention. However, gradient vanishing occurred due to very deep network layers. Olowolayemo et al. [11] proposed a mortality risk prediction model using fused CNN-extracted image features with structured health variables to generate risk scores. However, image-derived features lacked discriminative power due to poor feature correlation. Thilagavathi et al. [12] proposed multi-level convolutional filters and channel-wise attention. However, segmentation boundaries lacked precision when local features were weak. Agarwal and Arya [13] proposed CXRNet, a CNN enhanced with attention modules to focus on disease-relevant regions and suppress background noise. The feature maps showed redundancy due to a lack of feature selection.

Jacob and Lal [14] proposed C19XNet, a multiclass model that classified lungs using a tandem of parallel CNN paths and decision fusion. The model suffered slow convergence due to a late fusion strategy. Roy et al. [15] proposed a pooling-based Vision Group Geometry (VGG) Lite network tailored to detect outcomes; it replaced max-pooling with hybrid pooling to preserve spatial information.

The high-dimensional output led to increased classifier overhead. Alotaibi et al. [16] developed a CNN model with a modified ResNet-50. Their method achieved strong accuracy but required significant computing resources due to extensive convolutional layers.

Singh et al. [17] proposed Deep CP CXR, a deep learning model combining dual-path convolution modules for feature fusion. The overlapping feature boundaries led to classification ambiguity. Fu et al. [18] proposed LungMaxViT, an explainable hybrid Vision Transformer (ViT) structure integrating CNN initial blocks with Squeeze-and-Excitation (SE) modules to classify multiple diseases. The model still extracted noisy features due to a lack of explicit feature pruning. Padmavathi & Ganesan [19] proposed a ViT-based severity detection model optimized through metaheuristic techniques on multimodal COVID-19 images. Here, high performance came at the cost of increased compute complexity.

Pal et al. [20] proposed a comparative study exploring models, including ensemble methods. The feature-level misalignment reduced multiclass performance. Wang et al. [21] proposed TMscNet, a deep network with multiple information interaction layers to integrate spatial and contextual features. The model suffered from feature overlap due to the absence of effective feature selection. Ameta et al. [22] combined DenseNet 121, a densely connected CNN, with a Support Vector Machine (SVM), leveraging rich feature reuse and margin-based decision boundaries. The elevated feature dimensionality increased computational complexity. Islam [23] employed a genetic algorithm to optimize layer configurations and freezing strategies in CNNs for low-cost disease detection, targeting efficient model design. The model suffered inconsistent feature alignment across tuned layers. Kumar et al. [24] introduced a hybrid VGG16 enhanced with texture rectified cross attention to combine CNN and attention mechanisms; it emphasized both local textures and global context. The transformer extracted noisy texture features due to a lack of feature pruning. M. R et al. [25] used advanced InceptionV3 techniques and optimal weight initialization methods to classify COVID-19. The pretrained feature misalignment reduced classification consistency.

2.2. Related Work on CT Dataset

Balasamy and Seethalakshmi [26] proposed HCO-RLF, which fused recurrent learning via a hybrid optimization framework. The framework increased feature redundancy due to a lack of dimensionality control. Kordnoori et al. [27] designed a deep learning framework that processed raw CT scans end-to-end for accurate COVID-19 classification.

It demonstrated strong detection rates but suffered from over-parameterisation. Sahu and Kashyap [28] proposed FINE_DENSEIGANET, a hybrid model that integrated GAN-based data augmentation with DenseNet to improve CT classification. Here, feature noise persisted due to unrefined synthetic content. Alharbi and Ahmad [29] presented the DI-QL approach, which incorporated quantum gates (Hadamard and coupling). Here, unpruned features led to increased classifier variance.

Fathy and Abdel-Kader [30] developed a method that applied meta-heuristic feature selection on deep CT image descriptors before classification. The feature inconsistency reduced stability in classification performance. Pham et al. [31] proposed XCT-COVID, a deep transfer learning framework augmented with Explainable AI (XAI) tools to diagnose COVID-19. The XAI outputs highlighted broad areas, which reduced feature localization precision, affecting interpretability. Antunes et al. [32] developed CTCovid19, fine-tuning ResNet50 on CT scans with added classification layers. The fine-tuned network centralized on deep features without any feature selection, which increased model redundancy.

Kordnoori et al. [33] introduced the LungXpertAI network that jointly performed segmentation and COVID-19 classification. The joint training caused feature conflicts, which reduced classification accuracy due to competing task objectives. Rezvani et al. [34] proposed FusionLungNet, combining multi-scale feature fusion. The fused multi-scale pipelines generated excessive feature maps, significantly increasing computational overhead. Padmavathi & Ganesan [19] integrated a ViT model with metaheuristic optimizers to classify severity using multimodal COVID-19 images, enhancing feature interaction and adaptive weighting. The transformer architecture required high computational resources, which lowered processing efficiency.

Suseela and Parekh [35] proposed an Attention-CNN model that used pixel-level attention mechanisms on chest CT images to focus on COVID-impact regions and improve diagnostic performance. It extracted redundant attention features due to the absence of feature selection. Appati et al. [36] proposed a bootstrapped ViT-B/16 model, which applied ViT pretraining and bootstrap aggregation for SARS detection. The transformer model suffered from noisy feature interference due to the lack of dimensionality reduction. Chowa et al. [37] proposed a VGG19 model that combined dense connectivity with neuro-fuzzy reasoning to classify COVID-19, delivering adaptive rule-based predictions. It suffered from unrefined neuro-fuzzy features, causing classification inconsistency.

Tan, Nurlaila et al. [38] proposed a lightweight VGG16 model designed for lung lesion segmentation in COVID-19 CTs, balancing model depth and resolution for accurate lesion delineation. The segmentation exhibited weak feature localization for minor lesions. Singh and Retinderdeep [39] proposed a hybrid deep CNN capable of diagnosing COVID-19, with InceptionV3, which included a heat map for disease region explanation. The model's high feature dimensionality increased computational complexity.

2.3. Research Gaps

The research gaps identified from the research are as follows

- Most existing works lack a complete pipeline that combines all critical stages: preprocessing, feature extraction-selection, and classification. Instead, many focus only on one or two stages, limiting overall system performance. This affects robustness and end-to-end accuracy.
- Very few studies have explored COVID-19 detection using CT and CXR datasets together. Most research targeted either CT or CXR independently, missing the opportunity to develop a unified, multi-modal diagnostic framework that enhances reliability.
- Current segmentation methods often rely on pre-trained weights or manual tuning, without using nature-inspired optimization to fine-tune segmentation parameters. This results in suboptimal region separation, affecting downstream feature quality.
- Advanced feature selection approaches like IBBO are not integrated into existing pipelines. Most models depend on basic selection methods, which fail to reduce feature redundancy effectively or optimize classifier input for better accuracy.

2.4. Novel Contributions

To overcome those research gaps, the novel contributions of the work are defined as follows:

- A novel CT-CXR-Net was developed with various pipelined stages with both datasets to improve classification performance.
- An OU-Net is optimized using MGWO for better boundary extraction to achieve precise lung region segmentation.
- ResNet50 is integrated for both CT and CXR images to extract deep, informative features, improving pattern recognition.
- IBBO is used to enhance classification accuracy and select the most relevant features.
- A Ridge Classifier is deployed to ensure reliable and balanced diagnosis, effectively managing bias and variance in predictions.

3. Proposed CT-CXR -Net

The proposed CT-CXR-Net introduces a unique combination of advanced denoising, segmentation, feature extraction, and selection techniques, which have not been presented collectively in any existing surveys for COVID-19 detection. This approach addresses the critical drawbacks identified in current literature, such as a lack of robust noise removal in medical images, sub-optimal segmentation accuracy, inefficient feature reduction, and inconsistent classification performance. Figure 1 shows the proposed CT-CXR-Net system architecture. This method forms an entirely new diagnostic pipeline by integrating BM3D denoising, optimal U-Net segmentation with MGWO, ResNet50-based deep feature extraction, IBBO for feature selection, and a Ridge Classifier.

3.1. Step 1: Image Acquisition

The methodology begins with the collection of two separate datasets, one comprising CT images and the other containing CXR images. Both datasets are processed independently to evaluate the model's performance across different imaging modalities used for COVID-19 detection.

3.2. Step 2: BM3D Denoising

Medical images often suffer from high-frequency noise, which can distort important diagnostic features. To address this, BM3D denoising is applied as a preprocessing step. Thus, ensuring cleaner and more reliable inputs for further processing.

3.3. Step 3: Optimal U-Net Segmentation with MGWO

U-Net segmentation is employed following denoising. MGWO is integrated to fine-tune the U-Net loss function to enhance segmentation precision. MGWO dynamically adjusts the hyperparameters to minimize segmentation loss, ensuring that lung regions are accurately isolated, reducing false detection from irrelevant areas.

3.4. Step 4: ResNet50 Deep Feature Extraction

Once the lung regions are segmented, ResNet50, a powerful deep CNN, is utilized for feature extraction. ResNet50 captures intricate hierarchical features from both CT and CXR images, providing deep, discriminative representations crucial for differentiating COVID-19-infected cases from healthy ones.

3.5. Step 5: IBBO for Feature Selection

Deep features often contain redundant or irrelevant information, which can hinder classification performance. To overcome this, the IBBOs algorithm is applied to optimize feature selection. IBBO enhances traditional Brown Bear Optimization by improving exploration and exploitation balance, leading to the selection of the most relevant and non-redundant features while reducing computational overhead.

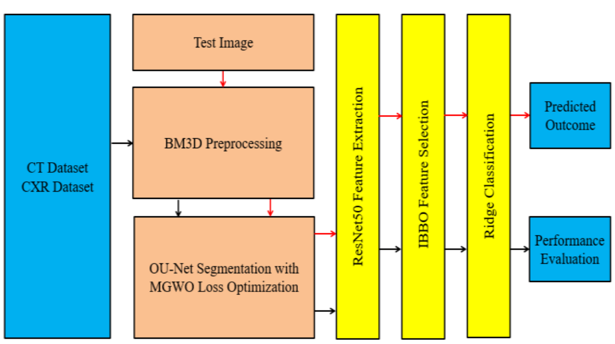

Fig. 1 Proposed CT-CXR-Net system architecture

3.6. Step 6: Ridge Classifier

The Ridge Classifier provides a stable and efficient classification mechanism, distinguishing COVID-19 positive and negative cases with high accuracy and generalizability across both CT and CXR datasets.

3.7. BM3D Denoising

Unlike basic filters that often blur edges and details, BM3D utilizes a combination of block matching, 3D collaborative filtering, and transform-domain thresholding, which allows for effective noise suppression while preserving structural information. Its unique approach of grouping similar patches across the image and applying joint filtering in a high-dimensional domain enhances sparsity, leading to superior denoising performance without sacrificing image quality. This results in clearer, more reliable medical images, essential for accurate diagnosis, especially when dealing with complex patterns in COVID-19-affected lung regions.

Figure 2 shows the proposed BM3D Denoising flowchart. Initially, the process begins by inputting either a CT or CXR image, which typically contains noise due to acquisition or transmission limitations, particularly in medical imaging. To ensure uniformity, the input image undergoes normalization to scale pixel intensity standards within a normal range, often [0, 1] or [0, 255]. Additionally, resizing performed to maintain consistent dimensions across the Dataset improves the efficiency of further processing steps. Let $\mathrm{I}_{input}(\mathrm{x}, \mathrm{y})$ is original data and $\mathrm{I}_{norm}(\mathrm{x}, \mathrm{y})$ is normalized data:

$$I_{input}(x, y) = \frac{I_{input}(x,y) - I_{min}}{I_{max} - I_{min}} \tag{1}$$

The normalized image is divided into small overlapping patches (blocks), and similar patches across the image are identified using a similarity metric (e.g., Euclidean distance).

These similar patches are grouped into a 3D array for joint processing, which increases the redundancy essential for effective denoising. Mathematically, for a reference patch P_r and candidate patch P_c:

$$D(P_r, P_c) = \sqrt{\sum_{i=1}^{N} \left(P_r(i) - P_c(i)\right)^2} \tag{2}$$

Here, D is the distance metric, and patches satisfying $D < \tau$ are grouped together. The grouped similar patches are transformed into a sparse domain using a combination of a 2D Discrete Cosine Transform (DCT) for spatial decorrelation within each patch and a 1D transform along the grouped dimension.

This enhances sparsity, making distinguishing noise from actual image content easier. For a patch $P(x, y)$, the 2D DCT is:

$$C(u, v) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} P(x, y). \cos\left(\frac{\pi(2x+1)u}{2N}\right) cos\left(\frac{\pi(2y+1)v}{2N}\right) \tag{3}$$

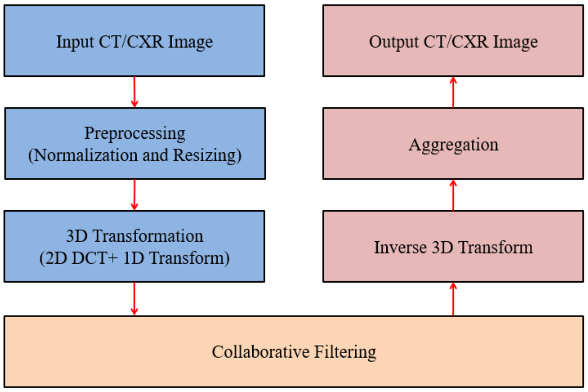

Fig. 2 Proposed BM3D denoising flowchart

In the transformed domain, collaborative filtering is applied to suppress noise. Two filtering methods are commonly used, such as hard threshold-based Wiener filtering. A statistical approach that weights the coefficients based on estimated signal and noise variances.

$$C'(u,v) = \begin{cases} C(u,v), & if\ |C(u,v)| \geq T \\ 0, & otherwise \end{cases} \quad (4)$$

Here, T is the hard threshold value, $C'(u,v)$ is the filtered outcome. The filtered patches are transformed back to the spatial domain using inverse 2D DCT and inverse 1D transform, reconstructing each patch in its denoised form. The inverse DCT:

$$P'(x,y) = \frac{1}{N}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} C'(u,v).\cos\left(\frac{\pi(2x+1)u}{2N}\right) cos\left(\frac{\pi(2y+1)v}{2N}\right) \quad (5)$$

Finally, overlapping denoised patches are aggregated by averaging to produce the final denoised image. This overlapping mechanism enhances reconstruction quality by reducing block artifacts. If multiple estimates $P'(x,y)$ Overlap at a pixel location, the aggregated output is:

$$I_{denoised}(x,y) = \frac{\sum_{i=1}^{M} w_i.\ P'(x,y)}{\sum_{i=1}^{M} w_i} \quad (6)$$

Where w_i There are weights based on the confidence of each patch estimate. The final denoised image $I_{denoised}(x,y)$ It is produced, significantly reducing noise while preserving critical diagnostic details, making it suitable for precise medical analysis in the subsequent steps of the proposed methodology.

3.8. Proposed OU-Net Segmentation

The proposed OU-Net offers significant advantages over conventional U-Net and other segmentation models by integrating MGWO for dynamic weight optimization, which is often absent in existing methods. Traditional U-Net relies on fixed or manually tuned weights, which can lead to sub-optimal segmentation, especially in complex medical images like CT or CXR scans with low contrast or noise. In contrast, OU-Net continuously adjusts the network weights through an iterative optimization process inspired by grey wolf hunting behavior, allowing the model to minimize segmentation loss more effectively.

This results in more precise lung region segmentation, better boundary detection, and improved overall performance, particularly for COVID-19 diagnosis. Additionally, OU-Net provides enhanced stability, faster convergence, and greater adaptability to varying datasets compared to static or purely deep learning-based approaches, making it highly reliable for real-time medical image segmentation tasks. Figure 3 shows the proposed OU-Net segmentation flowchart. The system begins with the input of preprocessed medical images (CT/CXR) of dimension 64×64×3, representing RGB or grayscale images resized for uniformity across the Dataset.

3.8.1. OU-Net Encoder

The OU-Net architecture initiates with the encoder path, where the input image undergoes successive convolutional layers and pooling operations to extract hierarchical features while reducing spatial dimensions.

$$F_{i,j}^{(l)} = \sigma\left(\sum_{m,n}^{N-1} I_{m,n}^{(l-1)}.K_{m,n} + b\right) \quad (7)$$

Here, $F_{i,j}^{(l)}$ Is the feature map at layer l, $I_{m,n}^{(l-1)}$ Is input from the previous layer, K is the convolutional kernel, b is the bias term, σ is the Rectified Linear Unit (ReLU). The output of the encoder captures essential features required for segmentation. At the bottleneck layer, the most abstract. This contains critical spatial and semantic information, but in a reduced form.

3.8.2. MGWO Initialization (Metaheuristic Weight Assignment)

Simultaneously, MGWO initializes a population of "wolves," where each wolf represents a possible set of U-Net weights.

MGWO mimics grey wolf hunting behavior with modified operators. Each wolf's fitness is estimated based on a validation set's segmentation performance (loss function).

3.8.3. Fitness Evaluation (Loss Calculation)

The fitness of each wolf (weight set) is assessed by calculating the segmentation dice loss.

$$L_{Dice} = 1 - \frac{2x|P \cap G|}{|P|+|G|} \quad (8)$$

Here, P is the predicted mask, G is the ground truth mask. Lower loss indicates better fitness, guiding the optimization process.

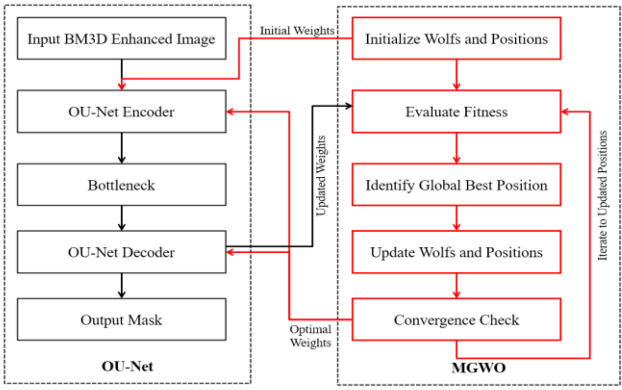

Fig. 3 Proposed OU-Net segmentation flowchart

3.8.4. Global Best Identification

MGWO identifies the global best position (i.e., the best-performing set of U-Net weights) among the current population, which represents the most optimal segmentation capability at that iteration.

3.8.5. Wolf Position Update (Weight Optimization)

The positions (weights) of the wolves are updated using MGWO-specific equations inspired by social hierarchy and hunting strategies. Modified Grey Wolf Optimization enhances the standard GWO with improved convergence behavior. The update equations:

$$D_\alpha = \mid C_1 \cdot X_\alpha - X \mid \tag{9}$$

$$X_1 = X_\alpha - A_1 \cdot D_\alpha \tag{10}$$

Here, X_α The position of the best wolf, X is the current wolf position, A_1, C_1 The coefficients controlling exploration and exploitation. These updates guide the weights toward globally optimal values.

3.8.6. Convergence Check

Upon termination, the optimal set of U-Net weights, determined through MGWO, is applied to the segmentation model. These weights yield the most accurate lung region masks with minimized loss, improving segmentation precision over standard U-Net training.

3.8.7. OU-Net Decoder

With optimized weights, the U-Net decoder performs upsampling and feature reconstruction, restoring the image to its original dimensions while generating an accurate segmentation mask highlighting the lung regions. This facilitates precise isolation of affected areas, critical for downstream COVID-19 classification.

3.9. ResNet50 Feature Extraction

ResNet50 offers clear advantages over traditional CNNs by introducing residual connections, which effectively solve the vanishing gradient problem that often limits the depth and performance of deep models.

Unlike standard CNNs, where increasing layers can degrade accuracy, ResNet50 allows for much deeper architectures by enabling shortcut paths that carry information forward, preserving both low-level and high-level features. This results in more stable training, faster convergence, and better generalization on complex tasks like medical image analysis. Additionally, its bottleneck design reduces computational complexity without compromising performance, making ResNet50 both efficient and highly effective for extracting detailed, discriminative features.

Figure 4 shows the proposed ResNet50 feature extraction flowchart. Channels are adjusted or replicated for grayscale medical images like CT or CXR scans to fit this dimension.

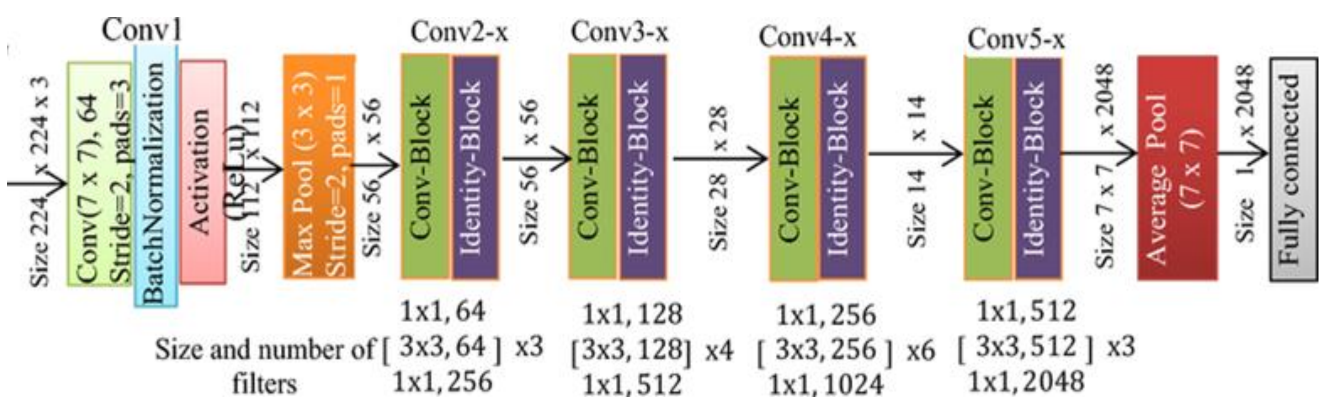

Fig. 4 Proposed ResNet50 feature extraction flowchart

3.9.1. Initial Convolution and Down-sampling

Mathematically, the convolution operation is:

$$F_{i,j,k} = \sum_{m,n,c}^{N-1} I_{m,n,c} \,.\, K_{i-m,j-n,c,k} + b_k \tag{11}$$

Here, $F_{i,j,k}$ Is the output feature map at position (i, j) and filter k, $I_{m,n,c}$ Is input data pixels, K is the convolution kernel, b_k It is a biased term. The Batch Normalization and ReLU are used to normalize activations and introduce non-linearity.

3.9.2. Max Pooling Layer

A 3×3 max pooling layer, while retaining the most prominent activations. This focuses on strong feature responses for downstream layers.

3.9.3. Conv2-x with Identity Block

This stage applies a combination of convolutional blocks and identity blocks, implementing skip connections and preventing degradation in deep networks. The residual mapping is expressed as:

$$H(x) = F(x, \{W_i\}) + x \tag{12}$$

Here, $F(x, \{W_i\})$ Contains non-linear transformations including convolutions, activations, and x is a shortcut connection (input). The Output size remains 56×56, with 64 feature channels.

3.9.4. Conv3-x with Identity Block

This block applies deeper convolutions with increasing filters (128 channels), and identity mappings to preserve gradient flow. Output dimensions are reduced to 28×28, capturing more abstract patterns and structural features.

3.9.5. Conv4-x with Identity Block

With additional convolutional and identity blocks, filters increase to 256, and spatial dimensions reduce to 14×14. This stage captures complex features such as shapes, patterns, and relevant high-level structures crucial for tasks like detecting lung anomalies in CT/CXR images.

3.9.6. Conv5-x with Identity Block

The deepest layer applies residual learning with 512 filters, reducing the feature maps to 7×7 spatial dimensions, encoding highly abstract, discriminative representations essential for classification or detection.

3.9.7. Global Average Pooling (GAP)

A GAP layer compresses the 7×7×2048 feature map into a 1×2048 vector by averaging each feature map:

$$GAP_k = \frac{1}{M*N} \sum_{i=1}^{N-1} \sum_{j=1}^{M-1} F_{i,j,k} \tag{13}$$

Here, $F_{i,j,k}$ Is the activation. The resulting 2048-dimensional feature vector is extracted as the deep feature representation of the input data. This rich feature set was highly discriminative and passed to the IBBO feature selection procedure.

3.10. IBBO Feature Selection

The IBBO feature selection method offers distinct advantages over traditional selection techniques by combining both exploration and exploitation in a more balanced, adaptive way. Unlike conventional methods that get trapped in local optima or depend heavily on random search, IBBO simulates brown bears' intelligent foraging and social behaviors to refine feature subsets dynamically. This results in better convergence speed, higher chances of finding globally optimal solutions, and reduced feature redundancy. By selecting only the most relevant and non-redundant features efficiently, IBBO enhances classification performance, lowers computational complexity, and significantly improves model generalization, especially in high-dimensional datasets like medical imaging or deep feature spaces.

Figure 5 shows the proposed IBBO feature selection flowchart. The process begins by defining the feature search space using deep features extracted from ResNet50. Assume there are N total features, and each potential feature subset is represented as a binary vector:

$$S = [s_1, s_2, s_3, \dots, s_N] \tag{14}$$

Here, $s_i = 1$ indicates the feature is selected, and $s_i = 0$ indicates exclusion.

3.10.1. Bear Population Initialization

A population of candidate solutions (bears) was primed, with each bear representing a different feature subset S. The population size is P, and each bear has its unique feature combination for evaluation.

3.10.2. Fitness Function Definition

A common fitness function balances classification accuracy and subset reduction:

$$F(S) = \alpha \times \left(1 - Acc(S)\right) + \beta \times \left(\frac{|S|}{N}\right) \tag{15}$$

Here, $Acc(S)$ is classification accuracy using subset S, $|S|$ The number of selected features, N is the total features, α, β are weights to balance accuracy and subset size.

3.10.3. Global Best and Personal Best Update

The algorithm tracks the global best-performing bear (best feature subset) and each bear's personal best position to guide the optimization process. Here,G_{best} is the best feature subset, P_{best} Is the personal best of bear i.

3.10.4. Apply IBBO Operators

Bears adjust their feature subset positions using two main behaviours. Foraging (Exploration), where bears randomly explore new regions of the search space.

Socialization (Exploitation), where bears imitate successful peers to refine feature selection. Position updates are modelled as:

$$S_i^{new} = S_i^{current} + \gamma \times (P_{best}^i - S_i^{current}) + \delta \times (G_{best}^i - S_i^{current}) \tag{16}$$

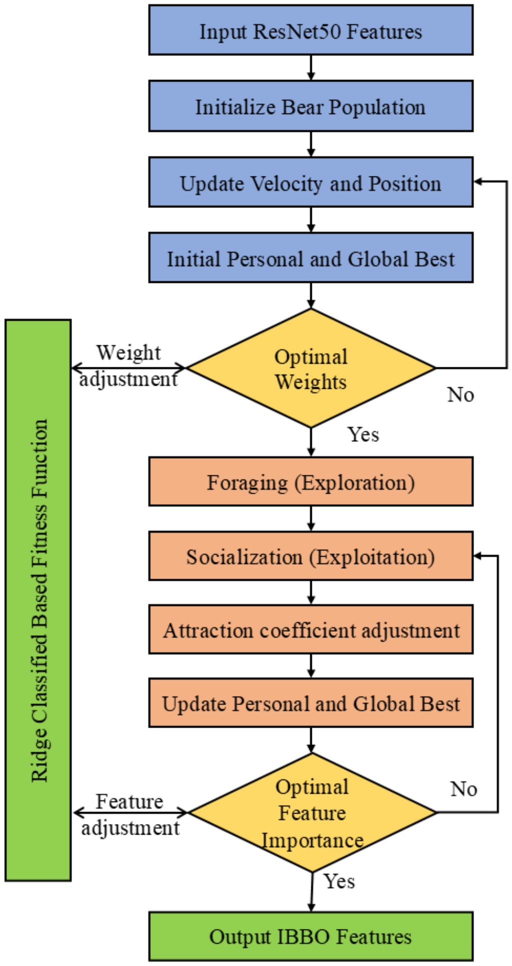

Fig. 5 Proposed IBBO feature selection flowchart

Here, γ, δ are random factors controlling exploration and exploitation. Binary positions are discretized using a transfer function (e.g., sigmoid) to ensure feature inclusion/exclusion:

$$S_i^{new} = \begin{cases} 1, & if\ rand() < \sigma(S_i^{new}) \\ 0, & otherwise \end{cases} \tag{17}$$

Here, σ is a sigmoid transfer function.

3.10.5. Iterative Optimization Loop

Repeat for a defined number of iterations or until convergence criteria are met (e.g., no significant fitness improvement). This ensures that bears continuously refine feature subsets to reach optimal solutions. Once maximum iterations are reached or convergence is achieved, the feature subset associated with the global best bear is selected for final model development.

3.11. Ridge Classifier

The Ridge Classifier offers a simple yet highly effective solution for handling high-dimensional data, especially when features are correlated or datasets have more features than samples. Figure 6 shows the proposed didge classifier flowchart. Unlike traditional classifiers that can easily overfit in such scenarios, this prevents the model from becoming overly complex. This leads to better generalization on unseen data. Additionally, it is computationally efficient, easy to implement, and particularly robust when dealing with multicollinearity or noisy features, for medical image classification, where interpretability and stability are crucial.

3.11.1. Initiation

The training phase begins with initializing the Ridge Classifier and providing the training dataset, consisting of **a** feature matrix $X_{train} \in R^{m \times n}$, label vector $y_{train} \in R^m$, here m is the number of samples, n is the number of features.

3.11.2. Set Regularization Parameter (α)

The regularization strength α is set to control the trade-off between minimizing classification loss and penalizing large coefficient values. Higher α reduces model complexity, enhancing generalization.

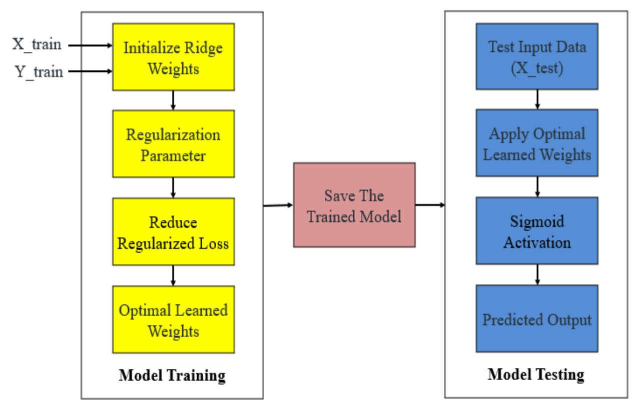

Fig. 6 Proposed ridge classifier flowchart

3.11.3. Training

The ridge classifier minimizes the L2-regularized loss function to find optimal model weights w. For binary classification, this often involves minimizing the following cost:

$$J(w) = L(y_{train}, X_{train} \cdot w) + \alpha |W|_2^2 \tag{18}$$

Here, L is hinge loss for classification, $|W|_2^2$ is the sum of squared model coefficients (L2 penalty). This step ensures both accurate classification and prevents overfitting by discouraging large weights.

3.11.4. Prediction Phase

Once the loss is minimized, the model outputs the learned coefficients w, which are the optimized weights assigned to each feature, reflecting their contribution to the prediction. Predictions are computed by applying the learned weights w to the test feature set:

$$z = X_{test} \cdot w \tag{19}$$

Here, z represents the raw prediction scores for each test sample. A decision function is applied to the prediction scores. For binary classification, this usually involves applying the sigmoid function:

$$\hat{y} = \begin{cases} 1, & if\ \sigma(z) > 0.5 \\ 0, & otherwise \end{cases} \tag{20}$$

Here, $\sigma(z)$ is the sigmoid activation. The predicted class labels $\hat{y}$ They are generated as the final output, indicating the model's classification decision for each test sample (e.g., COVID-19 positive or negative).

4. Results and Discussion

The section presents a comparative analysis of different methods evaluated on the same CT and CXR datasets. It highlights performance variations across models using standard metrics.

4.1. Datasets

4.1.1. CT Dataset

The CT-based COVID-19 dataset of a total of 1229 images of the COVID-19 class and 1229 cases of COVID-19 not being present in the downloaded data. To access the Dataset, follow this link https://www.kaggle.com/datasets/plameneduardo/sarscov2-ctscan-dataset?resource=download

4.1.2. CXR Dataset

A group of researchers from Qatar University, Doha, Qatar, and the University of Dhaka, Bangladesh, together with their partners in Pakistan and Malaysia, in association with medical doctors, have prepared a collection of subjects of CXR images of COVID-19 positive persons, as well as Normal and Viral Pneumonia images. The data on COVID-19, normal, and other infection datasets of the lungs are published in phases.

In this initial release, we have published 219 COVID-19, 1341 normal and 1345 viral pneumonia CXR images. In the first update, we have augmented the COVID-19 category up to 1200 CXR images. The 2 nd update uses 3616 COVID-19 positive cases, 10,192 Normal, 6012 Lung Opacity (COVID lung infection) and 1345 Viral Pneumonia images with their lung masks. https://www.kaggle.com/datasets/anasmohammedtahir/covidqu

4.2. Results on CT Dataset

This section gives the detailed results of the proposed CT-CXT-Net with various stages on the CT dataset, including preprocessing, segmentation, and classification stages.

4.2.1. Preprocessing Results

Figure 7 illustrates the effect of BM3D denoising along with various data augmentation operations on CT images. The first set shows how noisy input images are rotated before and after BM3D denoising, highlighting that the noise-free image preserves structural clarity even after rotation. The second set demonstrates contrast enhancement, where images after BM3D denoising undergo contrast adjustment, producing sharper visual outputs.

Similarly, the final set displays intensity adjustment applied to both noisy and denoised images, confirming that BM3D preprocessing provides a cleaner foundation, leading to clearer intensity variations. These examples collectively demonstrate that BM3D denoising improves image quality and ensures data augmentation operations produce meaningful and reliable outputs for subsequent processing stages.

Figure 8 demonstrates the effect of BM3D denoising combined with multiple augmentation techniques applied to CT images. The first column shows noisy input images, followed by their scaled output, where size adjustments are clearly visible with improved clarity after noise removal. The second set of images illustrates how BM3D-denoised outputs undergo scaling, ensuring that structural details remain sharp and noise-free even after resizing.

Similarly, motion blur adjustments are applied to both noisy and denoised images, confirming that BM3D preprocessing significantly improves visual consistency even under simulated distortions.

The final set highlights sharpening operations on noisy and denoised images, demonstrating that BM3D provides a cleaner base image, enhancing the effectiveness of sharpening, thereby preserving fine lung structures.

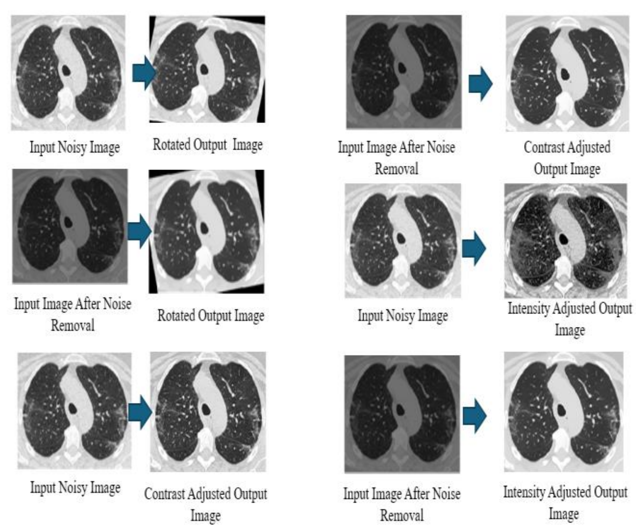

Fig. 7 BM3D denoising results with data augmentation techniques

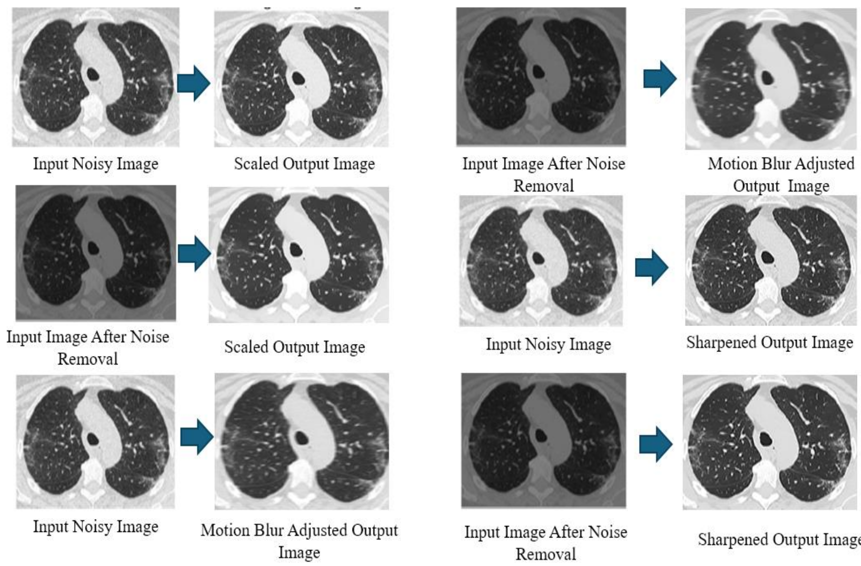

Fig. 8 BM3D denoising impact with various augmentation techniques on CT images

Table 1 presents the Peak Signal-to-Noise Ratio (PSNR) with Wavelet Thresholding (WT) [40] and BM3D algorithms under different preprocessing techniques. For COVID-19 images, the BM3D algorithm consistently provided higher PSNR values, indicating superior image quality, with 61.86487 for rotation, 76.26048 for scaling, 82.28839 for contrast adjustment, 82.28839 for intensity adjustment, 74.28114 for motion blur, 79.46884 for sharpening, and 82.28839 for flipping.

In comparison, WT [40] produced lower PSNR values with 55.60968 for rotation, 57.04691 for scaling, 51.05974 for contrast adjustment, 51.05966 for intensity adjustment, 57.04691 for motion blur, 57.04691 for sharpening, and 56.0276 for flipping. Similarly, for non-COVID-19 images, BM3D outperformed WT [40] across all methods, achieving 61.51998 for rotation, 78.69673 for scaling, 84.90672 for both contrast and intensity adjustment, 78.81164 for motion blur, 83.65212 for sharpening, and 84.90672 for flipping. In contrast, WT [40] resulted in comparatively lower PSNR values of 56.87786 for rotation, 58.84652 for scaling, 52.85734 for contrast adjustment, 52.85687 for intensity adjustment, 58.84652 for motion blur, 58.84652 for sharpening, and 57.84639 for flipping. The results clearly demonstrate the superior denoising and quality preservation capability of the BM3D algorithm in preprocessing tasks.

Table 2 presents the Structural Similarity Index Measure (SSIM) performance analysis under different preprocessing methods. The BM3D algorithm consistently achieved superior SSIM values compared to the WT method [40]. For COVID-19 images, the BM3D algorithm recorded 0.745489 for rotation, 0.843516 for scaling, 0.840268 for both contrast and intensity adjustment, 0.696439 for motion blur, 0.919483 for sharpening, and 0.840268 for flipping.

Table 1. PSNR performance analysis

Method	COVID-19		Non-COVID-19	
	WT [40]	BM3D	WT [40]	BM3D
Rotation	55.609	61.864	56.877	61.519
Scaling	57.046	76.260	58.846	78.696
Contrast Adjustment	51.059	82.288	52.857	84.906
Intensity Adjustment	51.059	82.288	52.856	84.906
Motion Blur	57.046	74.281	58.846	78.811
Sharpening	57.046	79.468	58.846	83.652
Flipping	56.027	82.288	57.846	84.906

In contrast, WT [40] resulted in lower SSIM values, with 0.232125 for rotation, 0.564695 for scaling, 0.000159 for contrast adjustment, 0.000155 for intensity adjustment, 0.564695 for motion blur, 0.564695 for sharpening, and 0.224782 for flipping.

Similarly, for Non-COVID-19 images, BM3D achieved SSIM values of 0.839128 for rotation, 0.909002 for scaling, 0.909821 for both contrast and intensity adjustment, 0.855923 for motion blur, 0.926469 for sharpening, and 0.909821 for flipping, while WT [40] produced 0.400267 for rotation, 0.693541 for scaling, 0.000569 for contrast adjustment, 0.000564 for intensity adjustment, 0.693541 for motion blur, 0.693541 for sharpening, and 0.396578 for flipping.

Table 3 shows the Mean Squared Error (MSE) performance, where lower values indicate better image quality. The BM3D algorithm consistently achieved the lowest MSE across all preprocessing methods. For COVID-19 cases, BM3D yielded 0.042325 for rotation, 0.001538 for scaling, 0.000384 for both contrast and intensity adjustment, 0.002426 for motion blur, 0.000735 for sharpening, and 0.000384 for flipping, significantly outperforming WT [40], which recorded 0.178695 for rotation, 0.128348 for scaling, 0.509455 for contrast adjustment, 0.509464 for intensity adjustment, 0.128348 for motion blur, 0.128348 for sharpening, and 0.162301 for flipping.

Similarly, for Non-COVID-19 cases, BM3D achieved 0.045823 for rotation, 0.000878 for scaling, 0.00021 for both contrast and intensity adjustment, 0.000855 for motion blur, 0.00028 for sharpening, and 0.00021 for flipping, while WT [40] recorded higher errors with 0.133443 for rotation, 0.084806 for scaling, 0.33678 for contrast adjustment, 0.336817 for intensity adjustment, 0.084806 for motion blur, 0.084806 for sharpening, and 0.106768 for flipping. These results clearly indicate that the BM3D algorithm provides significantly improved image quality compared to WT [40] under all preprocessing conditions.

Table 2. SSIM performance analysis

Method	COVID-19		Non-COVID-19	
	WT [40]	BM3D	WT [40]	BM3D
Rotation	0.2321	0.7454	0.4002	0.8391
Scaling	0.5646	0.8435	0.6935	0.9090
Contrast Adjustment	0.0001	0.8402	0.0005	0.9098
Intensity Adjustment	0.0001	0.8402	0.0005	0.9098
Motion Blur	0.5646	0.6964	0.6935	0.8559
Sharpening	0.5646	0.9194	0.6935	0.9264
Flipping	0.2247	0.8402	0.3965	0.9098

Table 3. MSE performance analysis

Method	COVID-19		Non-COVID-19	
	WT [40]	BM3D	WT [40]	BM3D
Rotation	0.1786	0.0423	0.1334	0.0458
Scaling	0.1283	0.0015	0.0848	0.00087
Contrast Adjustment	0.5094	0.00038	0.33678	0.00021
Intensity Adjustment	0.5094	0.00038	0.3368	0.00021
Motion Blur	0.1283	0.0024	0.0848	0.00085
Sharpening	0.1283	0.0007	0.0848	0.0002
Flipping	0.16230	0.0003	0.1067	0.0002

4.2.2. Segmentation Results

Figure 9 illustrates the segmentation outcomes for COVID-19 CT images. Figure 9 (a) shows the original CT images containing lung regions with visible infection areas. Figure 9 (b) presents segmentation results obtained using the Convolutional Auto Encoder and Decoder (CAED) [41] method, which partially captures infection regions but lacks structural precision. Figure 9 (c) shows the proposed OU-Net integrated with MGWO, achieving accurate infection boundary extraction with fewer artifacts and improved region continuity. Figure 10 depicts the segmentation outcomes for non-COVID-19 CT images. Figure 10 (a) shows the original lung CT images without COVID-19-specific infections. Figure 10 (b) demonstrates the CAED [41] segmentation, which produces noticeable over-segmentation with irregular boundaries. Figure 10 (c) displays the proposed OU-Net with the MGWO approach, delivering precise lung structure segmentation with enhanced smoothness and minimal false detections.

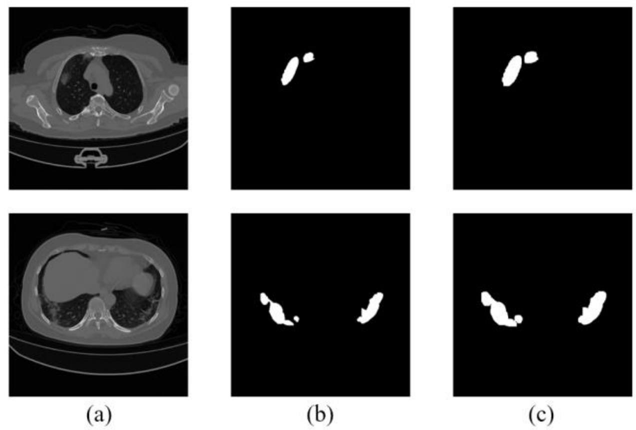

Fig. 9 Segmentation outcomes on COVID-19 CT images. (a)Original image, (b)CAED [41], and (c) Proposed OU-Net with MGWO

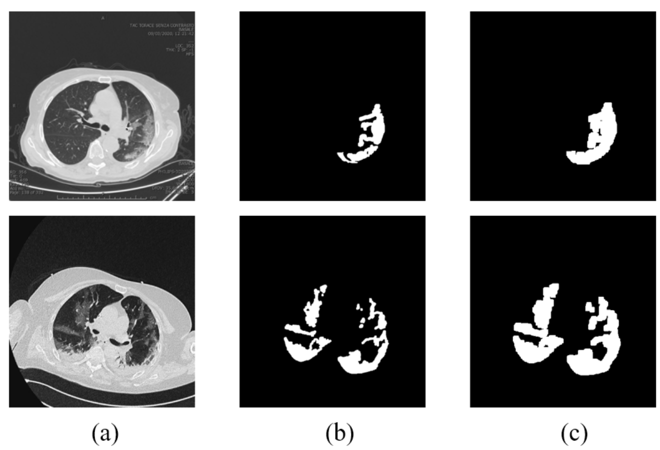

Fig. 10 Segmentation outcomes on Non-COVID-19 CT images. (a) Original image, (b) CAED [41], and (c) Proposed OU-Net with MGWO.

Table 4 presents the Segmentation Performance Analysis on CT datasets using existing CAED [41] and the proposed OU-Net with the MGWO method. For COVID-19 images, the proposed OU-Net with MGWO achieved significantly higher Accuracy of 0.9899 compared to 0.7333 with CAED [41], while Sensitivity improved from 0.4631 to 0.9899, and Specificity increased from 0.5207 to 1. The Precision and Recall also saw considerable enhancement, both reaching nearly 1 with the proposed method, compared to 0.8631 using CAED [41]. Similarly, the F1 score improved from 0.8631 to 0.9949, the Jaccard Index from 0.7271 to 0.9949, and the Dice Score from 0.7836 to 1. For Non-COVID-19 images, OU-Net with MGWO again outperformed CAED [41], with Accuracy increasing from 0.7333 to 0.9987, Sensitivity from 0.4990 to 0.9987, Specificity from 0.4037 to 1, Precision from 0.8990 to 1, Recall from 0.8990 to 0.9987, F1-Score from 0.8990 to 0.9993, Jaccard Index from 0.7309 to 0.9993, and Dice Score improving from 0.7836 to 1, clearly demonstrating the superiority of the proposed segmentation approach.

Table 4. Segmentation performance analysis on CT dataset

Metric	COVID-19 Images		Non-COVID-19 Images	
	CAED [41]	OU-Net with MGWO	CAED [41]	OU-Net with MGWO
Accuracy	0.733	0.9890	0.7333	0.9987
Sensitivity	0.463	0.989	0.4990	0.9987
Specificity	0.520	1	0.4037	1
Precision	0.863	1	0.8990	1
Recall	0.863	0.989	0.8990	0.9987
F1-Score	0.8631	0.994	0.8990	0.9993
Jaccard Index	0.72717	0.994	0.7309	0.999379
Dice Score	0.783	1	0.7836	1

4.2.3. Classification Results

Table 5 presents the overall Classification Performance Analysis on the CT dataset, comparing existing models with the proposed CT-CXR-Net. InceptionV3 [39] achieved 90.75% accuracy, 90.76% precision, 90.75% recall, and 90.75% F1-score. VGG16 [38] recorded slightly lower results with 89.23% accuracy, 89.78% precision, 89.23% recall, and 89.17% F1-score. VGG19 [37] performed similarly with 89.23% accuracy, 89.82% precision, 89.23% recall, and 89.16% F1-score. ResNet50 [31] outperformed these models with 96.54% accuracy, 96.61% precision, 96.54% recall, and 96.54% F1-score.

The proposed CT-CXR-Net delivered perfect performance with 100% across all metrics, showcasing its robustness for COVID-19 classification. Table 6 focuses on COVID-19 Class Performance Analysis using the CT dataset. InceptionV3 [39] achieved 91% across all metrics, while VGG16 [38] reached 85% accuracy and precision, 95% recall, and 90% F1-score. VGG19 [37] had the same accuracy and precision (85%), higher recall at 96%, and an F1-score of 90%. ResNet50 [31] provided better performance with 95% accuracy and precision, 98% recall, and 97% F1-score. Again, the proposed CT-CXR-Net outperformed all models, achieving a perfect 100% accuracy, precision, recall, and F1-score for COVID-19 cases.

Table 5. Classification performance analysis on CT dataset

Model	Acc.	Pre.	Recall	F1-Score
InceptionV3 [39]	0.9075	0.9076	0.9075	0.9075
VGG16 [38]	0.8923	0.8978	0.8923	0.8917
VGG19 [37]	0.8923	0.8982	0.8923	0.8916
ResNet50 [31]	0.9654	0.9661	0.9654	0.9654
Proposed CT-CXR-Net	1.000	1.000	1.000	1.000

Table 6. COVID-19 class performance analysis on CT dataset

Model	Acc.	Pre.	Recall	F1-Score
InceptionV3 [39]	0.91	0.91	0.91	0.91
VGG16 [38]	0.85	0.85	0.95	0.90
VGG19 [37]	0.85	0.85	0.96	0.90
ResNet50 [31]	0.95	0.95	0.98	0.97
Proposed CT-CXR-Net	1.00	1.00	1.00	1.00

Table 7 provides the Performance Analysis of the non-COVID-19 Class. InceptionV3 [39] achieved 90% performance and 91%. VGG16 [38] achieved an accuracy and precision of 94 percent as well as 83 percent recall with an 88 percent F1-score. The VGG19 [37] yielded an accuracy and precision of 95 percent, a recall of 82 percent, and an F1-score of 88 percent. A better outcome was revealed in ResNet50 [31] with 98 percent accuracy, the precision of 95 percent, the recall of 95 and an F1-score of 0.96. The tested CT-CXR-Net recorded an accurate performance, once again, recording 100% in performance.

Table 7. Non-COVID-19 class performance analysis on CT dataset

Model	Acc.	Pre.	Recall	F1-Score
InceptionV3 [39]	0.90	0.90	0.91	0.90
VGG16 [38]	0.94	0.94	0.83	0.88
VGG19 [37]	0.95	0.95	0.82	0.88
ResNet50 [31]	0.98	0.98	0.95	0.96
Proposed CT-CXR-Net	1.00	1.00	1.00	1.00

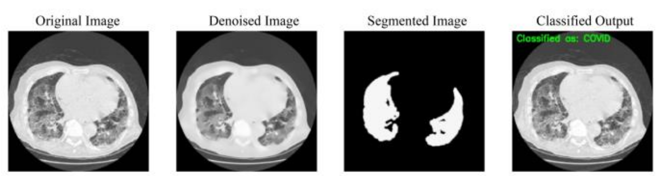

Fig. 11 Predicted outcome of COVID-19 by proposed CT-CXR-Net

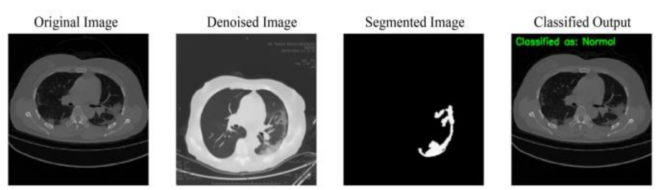

Fig. 12 Predicted outcome as non-COVID-19 by proposed CT-CXR-Net

Figure 11 displays the processing of a CT scan by the Proposed CT-CXR-Net, which ultimately classifies the image as COVID-19. The "Original Image" undergoes denoising and segmentation to highlight specific lung regions, leading to the "Classified Output" indicating the presence of COVID-19. Figure 12 illustrates the same processing pipeline by the Proposed CT-CXR-Net, but for a CT scan classified as non-COVID-19. The "Original Image" is denoised and segmented, and the "Classified Output" clearly states "Classified as: Normal," indicating the absence of COVID-19.

4.3. Results on CXR Dataset

This section gives the detailed results of the proposed CT-CXT-Net with various stages, including preprocessing, segmentation, and classification stages, on the CXR dataset.

4.3.1. Preprocessing Results

Figure 13 showcases the impact of different image preprocessing techniques (Intensity, Motion Blur, Sharpening, Flipping) applied, comparing the results of the Existing WT against a proposed BM3D method. The various transformations demonstrate how image characteristics are altered, aiming to improve clarity or generate diverse training data while highlighting the potential differences in output quality between the existing and proposed preprocessing approaches. Figure 14 illustrates the application of various image preprocessing techniques (Noisy, Rotate, Scaling, Contrast) to a non-COVID-19 image, comparing the outcomes of the existing WT with the proposed BM3D method. This comparison helps in evaluating how each preprocessing technique affects the visual quality and characteristics of normal lung images, and how the proposed method potentially offers superior or different results compared to the existing approach for non-COVID-19 cases.

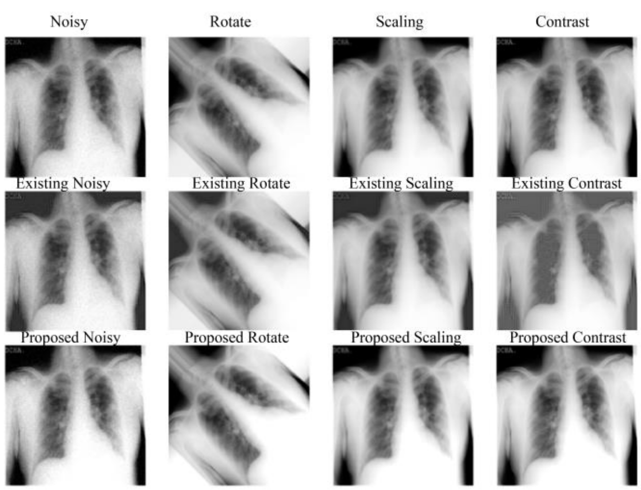

Fig. 13 Preprocessing results of existing WT and proposed BM3D on COVID-19 set

Table 8 presents the PSNR performance analysis on the CXR dataset under various preprocessing attacks, comparing WT [42] and BM3D methods. For Non-COVID-19 images, the BM3D approach consistently outperforms WT [42], achieving PSNR values of 66.57214 dB for noisy images, 59.08357 dB for rotation, 66.23095 dB for scaling, 66.26108 dB for contrast adjustment, 66.26106 dB for intensity adjustment, 66.18853 dB for motion blur, 66.65802 dB for sharpening, and 62.20782 dB for flipping, whereas WT [42] produced lower PSNR values ranging from 57.37731 dB to 57.82889 dB across these attacks.

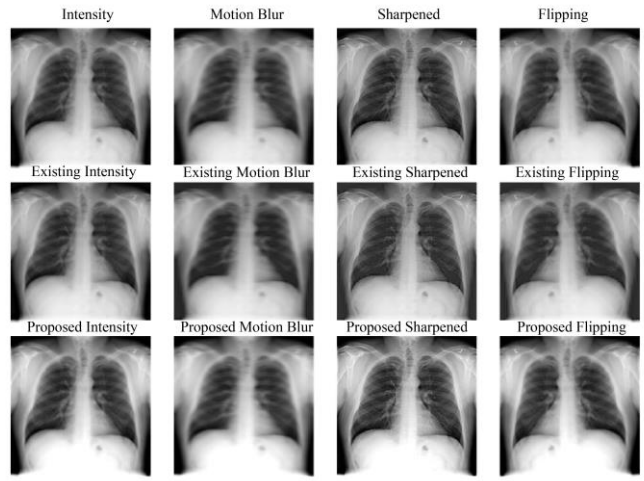

Fig. 14 Preprocessing results of existing WT and proposed BM3D on non-COVID-19 set

Table 8. PSNR performance analysis on CXR dataset

Attack	Non-COVID19		COVID-19	
	WT [42]	BM3D	WT [42]	BM3D
Noisy	57.8146	66.5721	56.3325	66.21806
Rotate	57.6524	59.0835	56.3115	60.91104
Scaling	57.8176	66.2309	56.33713	65.57614
Contrast	57.3773	66.2610	55.89617	65.74749
Intensity	57.6736	66.2610	56.20889	65.74746
Motion Blur	57.8072	66.1885	56.33586	65.57866
Sharpened	57.8288	66.6580	56.33684	65.79001
Flipping	57.7599	62.2078	56.3217	63.60607

Similarly, for COVID-19 images, BM3D consistently yielded better PSNR results with 66.21806 dB for noisy images, 60.91104 dB for rotation, 65.57614 dB for scaling, 65.74749 dB for contrast, 65.74746 dB for intensity, 65.57866 dB for motion blur, 65.79001 dB for sharpening, and 63.60607 dB for flipping, outperforming WT [42], which recorded values between 55.89617 dB to 56.33684 dB across all scenarios. These results confirm BM3D's superior noise reduction and image quality preservation.

Table 9 provides the SSIM performance analysis under different preprocessing attacks using WT [42] and BM3D techniques. For non-COVID-19 images, BM3D consistently delivers higher structural similarity, achieving SSIM values of

0.497122 for noisy images, 0.278917 for rotation, 0.915584 for scaling, 0.926807 for contrast adjustment, 0.926805 for intensity adjustment, 0.854739 for motion blur, 0.807974 for sharpening, and 0.392716 for flipping. In contrast, WT [42] produces significantly lower SSIM scores, ranging between 0.215127 and 0.267802.

Similarly, for COVID-19 images, BM3D outperforms WT [42] across all scenarios, with SSIM values of 0.249087 for noisy images, 0.445325 for rotation, 0.747633 for scaling, 0.747307 for contrast adjustment, 0.747305 for intensity adjustment, 0.743735 for motion blur, 0.557568 for sharpening, and 0.461381 for flipping. Meanwhile, WT [42] shows inferior SSIM results between 0.015612 and 0.099669. These outcomes demonstrate that BM3D substantially enhances image quality and structural similarity, making it more reliable under varied conditions.

Table 9. SSIM performance analysis on CXR dataset

Attack	Non-COVID19		COVID-19	
	WT [42]	BM3D	WT [42]	BM3D
Noisy	0.2628	0.4971	0.091431	0.249087
Rotate	0.2151	0.2789	0.099669	0.445325
Scaling	0.2642	0.9155	0.093201	0.747633
Contrast	0.2302	0.9268	0.015612	0.747307
Intensity	0.2610	0.9268	0.08236	0.747305
Motion Blur	0.2629	0.8547	0.09222	0.743735
Sharpened	0.2678	0.8079	0.095065	0.557568
Flipping	0.2519	0.3927	0.093258	0.461381

Table 10 presents the MSE Performance Analysis on the CXR, comparing WT and BM3D techniques under various attack conditions. For non-COVID-19 images, BM3D significantly reduces the error across all scenarios, with MSE values of 1.431747 for noisy images, 8.030138 for rotation, 1.548766 for scaling, 1.538056 for contrast adjustment, 1.538066 for intensity adjustment, 1.563967 for motion blur, 1.403715 for sharpening, and 3.911097 for flipping. In contrast, WT results in much higher MSE values ranging from 10.71991 to 11.8946.

Similarly, for COVID-19 images, BM3D demonstrates superior performance, with MSE values of 1.55337 for noisy images, 5.272009 for rotation, 1.800803 for scaling, 1.731137 for contrast adjustment, 1.731148 for intensity adjustment, 1.79976 for motion blur, 1.714271 for sharpening, and 2.834481 for flipping. Meanwhile, WT yields higher MSE values between 15.11359 and 16.72875, indicating greater reconstruction errors. These results confirm that BM3D offers more effective noise suppression and better preserves image quality than WT for COVID-19 and non-COVID-19 CXR images under diverse conditions.

Table 10. MSE performance analysis on CXR dataset

Attack	Non-COVID19		COVID-19	
	WT [42]	BM3D	WT [42]	BM3D
Noisy	10.755	1.431747	15.12941	1.55337
Rotate	11.164	8.030138	15.20299	5.272009
Scaling	10.747	1.548766	15.11359	1.800803
Contrast	11.894	1.538056	16.72875	1.731137
Intensity	11.109	1.538066	15.56651	1.731148
Motion Blur	10.773	1.563967	15.11802	1.79976
Sharpened	10.719	1.403715	15.11459	1.714271
Flipping	10.891	3.911097	15.16738	2.834481

4.3.2. Segmentation Results

Figure 15 illustrates the segmentation outcomes on COVID-19 CXR images. The original image was shown in Figure 15 (a), and Figure 15 (b) represents the ground truth segmented mask for comparison. Figure 15 (c) displays the output from CAED [43], which shows visible segmentation inaccuracies, whereas Figure 15 (d) demonstrates the superior segmentation by the proposed OU-Net with MGWO, offering clear lung region boundaries and higher precision. Figure 16 demonstrates the segmentation results for non-COVID-19 CXR images. The original input is presented in Figure 16 (a), with the segmented ground truth mask shown in Figure 16 (b). The segmentation by CAED [43] in Figure 16 (c) contains noticeable boundary irregularities, while the proposed OU-Net with MGWO in Figure 16 (d) achieves highly accurate, smooth, and well-defined lung region extraction.

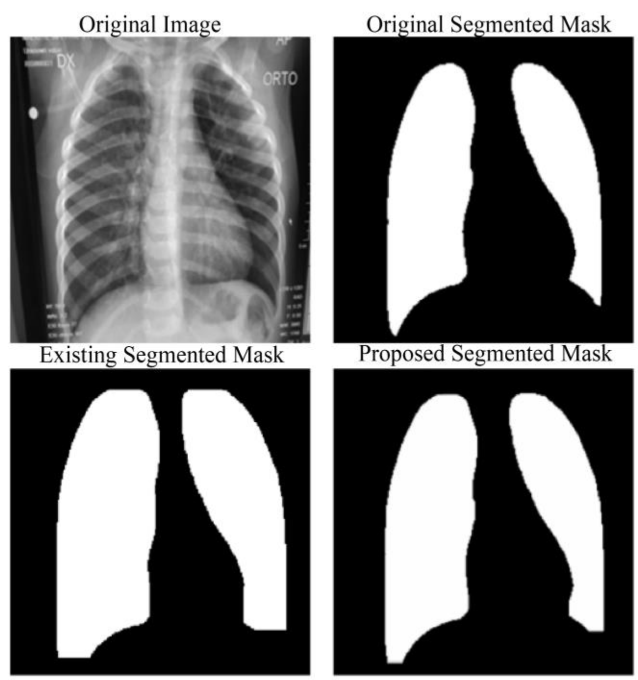

Fig. 15 Segmentation Outcomes on COVID-19 CXR Images. (a) Original Image, (b) Segmented Mask, (c) CAED [43], and (d) Proposed OU-Net with MGWO.

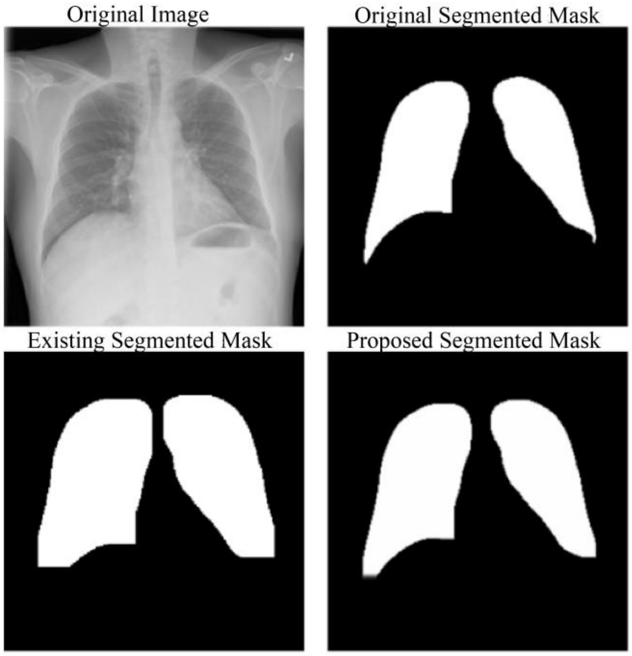

Fig. 16 Segmentation Outcomes on Non-COVID-19 CXR Images. (a) Original Image, (b) Segmented Mask, (c) CAED [43], and (d) Proposed OU-Net with MGWO.

Table 11. Segmentation performance analysis on CT dataset

Metric	Non-COVID19		COVID-19	
	CAED [43]	OU-Net with MGWO	CAED [43]	OU-Net with MGWO
Accuracy	0.8761	0.9859	0.8761	0.9859
Sensitivity	0.6205	0.9901	0.6205	0.9901
Specificity	0.95807	0.9852	0.9580	0.9852
Precision	0.9607	0.9867	0.9607	0.9867
Recall	0.91325	0.9047	0.9132	0.9047
F1-Score	0.9459	0.9830	0.9459	0.9830
Jaccard Index	0.91138	0.9955	0.9113	0.9955
Dice Score	0.9288	0.9818	0.9288	0.981

Table 11 presents the segmentation performance analysis on the CXR, comparing CAED [43] with the proposed OU-Net optimized by MGWO. For non-COVID-19 images, the proposed OU-Net with MGWO achieved a remarkable accuracy of 0.985961851, significantly outperforming CAED [43], which achieved 0.876179932. Sensitivity improved from 0.62057785 with CAED [43] to 0.990197178, reflecting better detection of lung regions. Specificity was 0.958072857 with CAED [43] and slightly decreased to 0.985264959 with OU-Net, indicating reliable exclusion of non-lung regions. Precision increased from 0.960761906 to 0.986770854, while Recall remained high at 0.904722103. The F1-Score rose from 0.945913308 to 0.983014536, demonstrating a balanced performance. The Jaccard Index improved from 0.911386396 to 0.995570809, and the Dice Score increased from 0.928866367 to 0.9818188, indicating excellent overlap between predicted and ground truth masks. Similar improvements were observed for COVID-19 images, with the same set of performance metrics showing consistent and substantial gains using the proposed OU-Net with MGWO compared to CAED [43], confirming the effectiveness of the proposed segmentation approach.

4.3.3. Classification Results

Table 12 presents the classification performance analysis on the CXR dataset, comparing the proposed CT-CXR-Net with existing models. The ResNet50 [13] model achieved 97.2% accuracy, 96.7% precision, 96.5% recall, and 96.5% F1-score, showing consistent performance across all metrics. The InceptionV3 [25] model obtained the highest accuracy among existing methods at 98.0%, with 98.4% precision, but a lower 90.8% recall and 90.8% F1-score, indicating missed positive cases. The VGG16 [24] and VGG19 [15] models both reached 97.5% accuracy, with 98.3% and 97.5% precision, but lower recall values of 89.2%, leading to 89.2% F1-scores, highlighting limitations in detecting positive cases. The proposed CT-CXR-Net significantly outperformed all existing methods, achieving an impressive 99.3% accuracy, 99.6% precision, 100% recall, and 100% F1-score, confirming its high reliability, minimal misclassification, and superior detection capability for COVID-19 in CXR images.

Table 12. Classification performance analysis on CXR dataset

Method	Acc.	Pre.	Recall	F1-score
ResNet50 [13]	0.972	0.967	0.965	0.965
InceptionV3 [25]	0.980	0.984	0.908	0.908
VGG16 [24]	0.975	0.983	0.892	0.892
VGG19 [15]	0.975	0.975	0.892	0.892
Proposed CT-CXR-Net	0.993	0.996	1.000	1.000

Table 13 illustrates the COVID-19 class-wise performance on the CXR data set of different deep learning models. The ResNet50 [13] had a detection sensitivity of 97.0%, precision of 96.5%, recall of 96.3%, and F1-score of 96.4 through balanced yet lower detection sensitivity. The InceptionV3 [25] model shows a better 97.5% accuracy with 97.8% precision, yet poor in recall (89.5%) and F1-Score (93.5%), which shows that the model tends to overlook the positive COVID-19 cases. On the same note, VGG16 [24] yielded 97.0 accuracy and precision but had a lower recall of 88.0 in addition to a 92.7 F1-score statistic, indicating the loss of Sensitivity. The VGG19 [15] model returned 97.2 accuracy, 97.3 precision, 88.5 recall and 92.7 F1-score with similar detection limitations to those of the other models. Comparatively, the given CT-CXR-Net report shows an accuracy of 99.0 percent, precision of 99.5 percent, recall of 100 percent and F1-score of 99.7 percent, which means high reliability of the COVID-19 cases and low rates of false negative specifications and the overall rates of classification with high values of 99.7 percent.

Table 14 represents a non-COVID-19 class performance analysis of the CXR dataset analysing various models. ResNet50 [13] scored 97.5 accuracy, 97.0 precision, 96.8 recall, and 96.9 F1-score, which are seemingly constant but at a slightly moderate level of performance. InceptionV3 [25] achieved an accuracy of 98.5%, precision of 99.0%, recall of 92.0%, and F1-Score of 95.4% percent, indicating a high precision but failing to capture all the negative cases. Equally, VGG16 [24] had an accuracy rate of 98.0%, precision of 98.5%, recall of 90.5% and F1-score of 94.3%, and VGG19 [15] had 97.8% accuracy, 97.8% precision, 90.0% recall and 93.7% scores due to a recall and overall classification deficiency. The suggested CT-CXR-Net model was able to outperform current approaches leading to an accuracy of 99.5 per cent, precision of 99.8 per cent, recall of 100 per cent, and F1-score of 99.9 per cent, guaranteeing a high level of performance in detecting non-COVID-19 sufficiently with few false positives and false negatives, thus, making it highly effective during clinical screening.

Table 13. COVID-19 class performance analysis on CXR dataset

Method	Acc.	Pre.	Recall	F1-score
ResNet50 [13]	0.970	0.965	0.963	0.964
InceptionV3 [25]	0.975	0.978	0.895	0.935
VGG16 [24]	0.970	0.980	0.880	0.927
VGG19 [15]	0.972	0.973	0.885	0.927
Proposed CT-CXR-Net	0.990	0.995	1.000	0.997

Table 14. Non-COVID-19 class performance analysis on CXR dataset

Method	Acc.	Pre.	Recall	F1-score
ResNet50 [13]	0.975	0.970	0.968	0.969
InceptionV3 [25]	0.985	0.990	0.920	0.954
VGG16 [24]	0.980	0.985	0.905	0.943
VGG19 [15]	0.978	0.978	0.900	0.937
Proposed CT-CXR-Net	0.995	0.998	1.000	0.999

Figure 17 illustrates the process of classifying a CXR as COVID-19 using CT-CXR-Net. The "Original Image" undergoes denoising to reduce noise, followed by segmentation to isolate the lung regions. The final "Classified Output" then indicates "Classified as: COVID" based on these processed images in CXR scans. Figure 18 demonstrates the classification of a CXR as non-COVID-19 by the Proposed CT-CXR-Net. Like the COVID-19 case, the "Original Image" is subjected to denoising and segmentation to highlight relevant areas. The "Classified Output" clearly displays "Classified as: Normal," signifying the model's capacity to identify non-COVID-19 cases and thus differentiate them from infected lungs on CXR images.

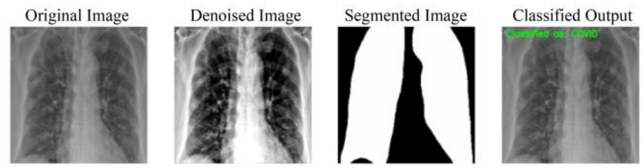

Fig. 17 Predicted outcome on CXR as COVID-19 by proposed CT-CXR-Net

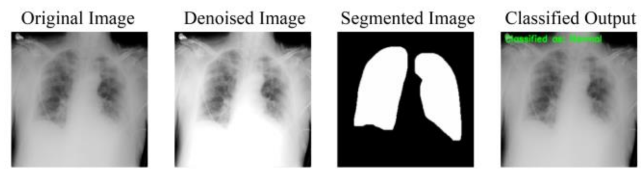

Fig. 18 Predicted outcome on CXR as non-COVID-19 by proposed CT-CXR-Net

5. Conclusion

The proposed CT-CXR-Net method integrates BM3D denoising, OU-Net segmentation with MGWO optimization, ResNet50 feature extraction, IBBO feature selection, and Ridge classification to achieve highly accurate COVID-19 and non-COVID-19 diagnosis on both CT and CXR datasets. The proposed CT-CXR-Net achieved an average improvement of 9.54% in accuracy, 9.52% in precision, 9.57% in recall, and 9.58% in F1-score over existing models on the CT dataset. The proposed CT-CXR-Net shows notable improvements over existing methods with an average increase of 2.37% in accuracy, 2.14% in precision, 6.75% in recall, and 5.25% in F1-score compared to the best-performing existing models on the CXR dataset. In the future, this framework was extended to real-time clinical environments, integrated with portable diagnostic devices, and adapted for early detection of other respiratory diseases such as pneumonia, lung cancer, or emerging viral infections, while exploring lightweight model compression and cross-hospital validation to enhance generalization and practical deployment.

References

[1] Devanshi Mallick et al., "Classifying Chest X-Rays for COVID-19 through Transfer Learning: A Systematic Review," *Multimedia Tools and Applications*, vol. 84, pp. 689-748, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[2] Warveen Merza Eido, and Hajar Maseeh Yasin, "Pneumonia and COVID-19 Classification and Detection Based on Convolutional Neural Network: A Review," *Asian Journal of Research in Computer Science*, vol. 18, no. 1, pp. 174-183, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[3] Yun-Chi Lin, and Yu-Hua Dean Fang, "Classification of the ICU Admission for COVID-19 Patients with Transfer Learning Models Using Chest X-Ray Images," *Diagnostics*, vol. 15, no. 7, pp. 1-17, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[4] Reenu Rajpoot et al., "Quantitative Assessment of XAI Methods for COVID-19 Detection: A Comparative Approach," *SN Computer Science*, vol. 6, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[5] Sushma Jaiswal, and Priyanka Gupta, "Ensemble Based Model for Diabetes Prediction and COVID-19 Mortality Risk Assessment in Diabetic Patients," *International Journal of Computer Engineering in Research Trends*, vol. 10, no. 3, pp. 99-106, 2023. [CrossRef] [Publisher Link]

[6] B. Narendra Kumar Rao, B. Bhaskar Kumar Rao, and Nagendra Panini Challa, *Harnessing Deep Learning for the Early Detection of COVID-19 using Chest X-Rays*, 1st ed., Big Data Analytics and Intelligent Applications for Smart and Secure Healthcare Services, CRC Press, pp. 1-14, 2025. [Google Scholar] [Publisher Link]

[7] Arthi Devarani Paulretnam, Inbamalar Tharcis Mariapushpum, and Swetha Ramamoorthi, "A Real Time Diagnosis of Lung Infection Using X-Ray Images," *AIP Conference Proceedings*, vol. 3159, no. 1, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[8] Sihan Li, and Juhua Huang, "ResGDANet: An Efficient Residual Group Attention Neural Network for Medical Image Classification," *Applied Sciences*, vol. 15, no. 5, pp. 1-20, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[9] Fatimah Alhafiz, and Abdullah Basuhail, "The Data Heterogeneity Issue Regarding COVID-19 Lung Imaging in Federated Learning: An Experimental Study," *Big Data and Cognitive Computing*, vol. 9, no. 1, pp. 1-24, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[10] Vijaya Patnaik, Monalisa Mohanty, and Asit Kumar Subudhi, "Classification of COVID-19 from Chest X-Ray Images Using ReSqNet," *Biomedical Engineering: Applications, Basis and Communications*, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[11] Akeem Olowolayemo et al., "Covid-19 Mortality Risk Prediction Using Small Dataset of Chest X-Ray Images," *Artificial Intelligence and Applications*, vol. 3, no. 1, pp. 94-109, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[12] P. Thilagavathi et al., "An Effective COVID-19 Classification in X-Ray Images Using a New Deep Learning Framework," *Journal of X-Ray Science and Technology*, vol. 33, no. 2, pp. 297-316, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[13] Saurabh Agarwal, and K.V. Arya, "CXRNet: CNN-Attention Based CXR Image Classifier," *Expert Systems*, vol. 42, no. 1, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[14] Basil Jacob, and Anisha M. Lal, "C19XNet: A Novel Multi-Classification Model for COVID-19 Detection in Chest X-Rays," *Fifth Congress on Intelligent Systems*, pp. 347-364, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[15] Santanu Roy et al., "Novel Pooling-Based VGG-Lite for Pneumonia and Covid-19 Detection from Imbalanced Chest X-Ray Datasets," *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[16] Reem Alotaibi et al., "Automated Diagnosis of COVID-19 Using Chest X-Ray Image Processing by a Convolutional Neural Network," *International Journal of Computer Mathematics*, vol. 102, no. 2, pp. 224-244, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[17] Kuljeet Singh et al., "Deep CP-CXR: A Deep Learning Model for Classification of Covid-19 and Pneumonia Disease Using Chest X-Ray Images," *Annals of Data Science*, pp. 1-24, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[18] Xiaoyang Fu et al., "Explainable Hybrid Transformer for Multi-Classification of Lung Disease Using Chest X-Rays," *Scientific Reports*, vol. 15, pp. 1-19, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[19] V. Padmavathi, and Kavitha Ganesan, "Metaheuristic Optimizers Integrated with Vision Transformer Model for Severity Detection and Classification via Multimodal COVID-19 Images," *Scientific Reports*, vol. 15, pp. 1-19, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[20] Madhumita Pal et al., "A Comparative Analysis of the Binary and Multiclass Classified Chest X-Ray Images of Pneumonia and COVID-19 with ML and DL Models," *Open Medicine*, vol. 20, pp. 1-16, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[21] Xicheng Wang et al., "TMscNet: A Model with Multiple Information Interaction for COVID-19 X-Ray Classification," *International Journal of Machine Learning and Cybernetics*, pp. 1-18, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[22] Gaurav Kumar Ameta et al., "Automated X-Ray Image Classification for COVID-19 Detection Using DenseNet-121 and SVM," *2025 International Conference on Automation and Computation (AUTOCOM)*, Dehradun, India, pp. 307-312, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[23] Md. Jahidul Islam, "Genetic Optimization-Based Layers Tuning and Freezing in Deep CNN for Low-Cost Disease Detection Using Chest X-Rays," *SN Computer Science*, vol. 6, pp. 1-19, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[24] Sunil Kumar, and Harish Kumar, "Efficient-VGG16: A Novel Ensemble Method for the Classification of COVID-19 X-Ray Images in Contrast to Machine and Transfer Learning," *Procedia Computer Science*, vol. 235, pp. 1289-1299, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[25] R. Mahaveerakannan, and K. Sai Pranay, "Automated COVID 19 Detection using InceptionV3 Algorithm to Enhance the Accuracy and Comparing with Shufflenet Algorithm," *2025 6th International Conference on Mobile Computing and Sustainable Informatics (ICMCSI)*, Goathgaun, Nepal, pp. 1472-1478, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[26] K. Balasamy, and V. Seethalakshmi, "HCO-RLF: Hybrid Classification Optimization Using Recurrent Learning and Fuzzy for COVID-19 Detection on CT Images," *Biomedical Signal Processing and Control*, vol. 100, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[27] Shirin Kordnoori et al., "A Deep Learning Framework for Accurate COVID-19 Classification in CT-Scan Images," *Machine Learning with Applications*, vol. 19, pp. 1-6, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[28] Hemlata P. Sahu, and Ramgopal Kashyap, "Fine_Denseiganet: Automatic Medical Image Classification in Chest CT Scan using Hybrid Deep Learning Framework," *International Journal of Image and Graphics*, vol. 25, no. 1, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[29] Meshal Alharbi, and Sultan Ahmad, "Enhancing COVID-19 Detection Using CT-Scan Image Analysis and Disease Classification: The DI-QL Approach," *Health and Technology*, vol. 15, pp. 477-488, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[30] Abdelghany Fathy, Hatem Abdel- Kader, and Amira Abdelatey, "COVID-19 Detection Based on CT Scan Using Meta-Heuristic Feature Selection Method," *International Journal of Computers and Information*, vol. 12, no. 1, pp. 24-42, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[31] Nhat Truong Pham et al., "Leveraging Deep Transfer Learning and Explainable AI for Accurate COVID-19 Diagnosis: Insights from A Multi-National Chest CT Scan Study," *Computers in Biology and Medicine*, vol. 185, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[32] Carlos Antunes, João Rodrigues, and António Cunha, "CTCOVID19: Automatic COVID-19 Model for Computed Tomography Scans Using Deep Learning," *Intelligence-Based Medicine*, vol. 11, pp. 1-10, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[33] Shirin Kordnoori et al., "LungXpertAI: A Deep Multi-Task Learning Model for Chest CT Scan Analysis and COVID-19 Detection," *Biomedical Signal Processing and Control*, vol. 99, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[34] Sadjad Rezvani et al., "FusionLungNet: Multi-Scale Fusion Convolution with Refinement Network for Lung CT Image Segmentation," *Biomedical Signal Processing and Control*, vol. 107, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[35] Suba Suseela, and Nita Parekh, "From Pixels to Prognosis: Attention-CNN Model for COVID-19 Diagnosis Using Chest CT Images," *IET Image Processing*, vol. 19, no. 1, pp. 1-17, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[36] Justice Kwame Appati et al., "SARS Detection in Chest CT Scan Images Using the Bootstrapped ViT-B/16 Model," *Iran Journal of Computer Science*, vol. 8, pp. 537-551, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[37] Sadia Sultana Chowa et al., "An Automated Privacy-Preserving Self-Supervised Classification of COVID-19 from Lung CT Scan Images Minimizing the Requirements of Large Data Annotation," *Scientific Reports*, vol. 15, no. 1, pp. 1-20, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[38] Nurlaila H. Tan, and Idam Arif, "Multiclass Classification of Covid-19 CT Scan Images With VGG-16 Architecture Using Transfer Learning System," *Indonesian Journal of Physics*, vol. 35, no. 1, pp. 21-26, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[39] Retinderdeep Singh et al., "High-Accuracy Detection of Cancer, COVID-19, and Tuberculosis in Chest CT Scans Using Fine-Tuned InceptionV3," *2024 IEEE 3rd World Conference on Applied Intelligence and Computing (AIC)*, Gwalior, India, pp. 987-993, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[40] Murat Alparslan Gungor, "A Comparative Study on Wavelet Denoising for High Noisy CT Images of COVID-19 Disease," *Optik*, vol. 235, pp. 1-7, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[41] Kiri S. Newson, David M. Benoit, and Andrew W. Beavis, "Encoder-Decoder Convolutional Neural Network for Simple CT Segmentation of COVID-19 Infected Lungs," *PeerJ Computer Science*, vol. 10, pp. 1-17, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[42] Francesco Prinzi et al., "Impact of Wavelet Kernels on Predictive Capability of Radiomic Features: A Case Study on COVID-19 Chest X-Ray Images," *Journal of Imaging*, vol. 9, no. 2, pp. 1-18, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[43] Putri Syafira, Novanto Yudistira, and Diva Kurnianingtyas, "Chest X-Ray Images Clustering Using Convolutional Autoencoder for Lung Disease Detection," *JOIV: International Journal on Informatics Visualization*, vol. 9, no. 2, pp. 838-844, 2025. [CrossRef] [Google Scholar] [Publisher Link]