Improving Biomedical Image Pattern Identification by Deep B4GraftingNet: Application to Pneumonia Detection

IET Image Processing

WILEY

The Institution of Engineering and Technology

ORIGINAL RESEARCH OPEN ACCESS

Improving Biomedical Image Pattern Identification by Deep B4-GraftingNet: Application to Pneumonia Detection

Syed Adil Hussain Shah[1,2] | Syed Taimoor Hussain Shah[1] | Abdul Muiz Fayyaz[3] | Syed Baqir Hussain Shah[4] | Mussarat Yasmin[4] | Mudassar Raza[5,6] | Angelo Di Terlizzi[2] | Marco Agostino Deriu[1]

[1]PolitoBIOMed Lab, Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Turin, Italy | [2]Department of Research and Development (R&D), GPI SpA, Trento, Italy | [3]Computer Information Science Department, Universiti Teknologi PETRONAS, Seri Iskandar, Perak, Malaysia | [4]Department of Computer Science, COMSATS University Islamabad (CUI), Wah, Pakistan | [5]Department of Computer Science, HITEC, University of Taxila, Taxila, Pakistan | [6]Department of Computer Science, Namal University Mianwali, Mianwali, Pakistan

Correspondence: Syed Adil Hussain Shah (syedadilhussain.shah@gpi.it)

Received: 25 September 2024 | **Revised:** 19 March 2025 | **Accepted:** 24 March 2025

Funding: The present research work has been developed as part of the PARENT project, funded by the European Union's Horizon 2020 Research and Innovation Program under the Marie Sklodowska-Curie Innovative Training Network 2020, Grant Agreement No. 956394. Additionally, this work has received support from Project D34 Health (PNC 0000001, CUP B83C22006120001) under the National Plan for Complementary Investments to the National Recovery and Resilience Plan (PNRR), funded by the European Union – NextGenerationEU.

ABSTRACT

VGG-16 and Inception are widely used CNN architectures for image classification, but they face challenges in target categorization. This study introduces B4-GraftingNet, a novel deep learning model that integrates VGG-16's hierarchical feature extraction with Inception's diversified receptive field strategy. The model is trained on the OCT-CXR dataset and evaluated on the NIH-CXR dataset to ensure robust generalization. Unlike conventional approaches, B4-GraftingNet incorporates binary particle swarm optimization (BPSO) for feature selection and grad-CAM for interpretability. Additionally, deep feature extraction is performed, and multiple machine learning classifiers (SVM, KNN, random forest, naïve Bayes) are evaluated to determine the optimal feature representation. The model achieves 94.01% accuracy, 94.22% sensitivity, 93.36% specificity, and 95.18% F1-score on OCT-CXR and maintains 87.34% accuracy on NIH-CXR despite not being trained on it. These results highlight the model's superior classification performance, feature adaptability, and potential for real-world deployment in both medical and general image classification tasks.

1 | Introduction

In recent years, image classification and object detection have witnessed significant advancements in deep learning architectures [1–3]. These improvements stem from enhanced hardware capabilities, larger datasets, and the development of more sophisticated computational models. Over time, these advancements have contributed to refined algorithms and improved architectures, allowing for more efficient and accurate feature learning [4]. Among the many deep learning frameworks, convolutional neural networks (CNNs) have proven to be particularly influential. Researchers have introduced various CNN architectures, including VGG [5], GoogleNet [6], ResNet [7], and Transformers [8], each with distinct characteristics and strengths. For instance, VGG-16 is known for its deep hierarchical feature extraction using small convolutional filters, while Inception excels in classification tasks by employing parallel convolutional filters within each inception block. The impact of Inception extends beyond its own architecture, as its core design principles have been integrated into several state-of-the-art CNN models [6, 9, 10].

Syed Adil Hussain Shah and Syed Taimoor Hussain Shah contributed equally to this study.

Summary Points

What was already known on the topic?

- VGG-16 and Inception models are well-known for their ability to identify objects and capture spatial information within images. However, these models often face challenges in delivering consistent and optimal performance across complex datasets.

What this study added to the knowledge?

1. Development of B4-GraftingNet: Introduced a novel 87-layer CNN architecture, B4-GraftingNet, which combines the strengths of VGG-16 and Inception patterns to improve target categorization and spatial information computation.
2. Implementation of GraftingNet Technique: Incorporated a GraftingNet approach inspired by Inception modules to efficiently propagate diverse receptive field information throughout the CNN network.
3. Feature Extraction Across Datasets: Trained the proposed model on the OCT-CXR dataset and extracted features from both the OCT-CXR and NIH clinical (holdout) datasets for downstream processing.
4. Feature Optimization with BPSO: Utilized binary particle swarm optimization (BPSO) to identify and select the most optimized features for improved classification performance.
5. Classifier Evaluation: Evaluated the performance of various classifiers integrated into the proposed pipeline, demonstrating the effectiveness of the technique on the OCT-CXR test dataset and validating the results on the NIH clinical dataset through a holdout evaluation.
6. Explainable AI (XAI): Applied features visualization and grad-CAM to identify critical regions in images, ensuring the robustness and interpretability of the B4-GraftingNet model for clinical applications.

Building on these architectures, this paper proposes a novel image classification model, B4-GraftingNet, which synergistically combines the hierarchical feature extraction of VGG-16 with the diversified receptive field strategy of Inception. The proposed model enhances feature learning and class discrimination, making it highly efficient for various classification tasks. While the model is not restricted to a specific domain, it is particularly useful for complex medical imaging applications, where subtle feature distinctions are harder to classify compared to large-scale object detection tasks. As a case study, we evaluate the model on pneumonia classification, a challenging medical imaging task that requires high precision and sensitivity due to the intricacies of lung diseases.

Pneumonia is a severe lung infection, primarily caused by bacteria, viruses, or fungi, and is one of the leading causes of global mortality. In 2019, it was responsible for 2.5 million deaths, including 672,000 children, equating to one child under five dying every 47 s. Among neonates and premature babies, particularly those born before 37 weeks of gestation, the risk is three to ten times higher due to underdeveloped lung structures and weak immune systems. Pneumonia accounts for 15% of all neonatal deaths, and its complications can lead to impaired brain development, vision and hearing impairments, and chronic lung diseases. Early diagnosis and timely medical intervention are essential in reducing pneumonia-related fatalities [11, 12]. The disease remains a global health challenge, with 14% of all disease-related deaths in children under five attributed to pneumonia [13]. Tuberculosis (TB), another respiratory disease, caused 230,000 deaths in 2020, 80% of which were among children under five [14]. Despite improvements in healthcare, pneumonia continues to pose serious health risks, particularly in low- and middle-income countries, where limited access to early diagnosis and treatment exacerbates mortality rates. Reports from the Western Pacific region indicated that pneumonia accounted for 11% of annual deaths in children under five in 2011, with the highest fatality rates recorded in Pakistan, India, Nigeria, Ethiopia, and the Republic of Congo in 2017 [15–17].

The role of information technology (IT) and artificial intelligence (AI) in healthcare has become increasingly critical in improving diagnostic accuracy and efficiency [18]. Deep learning-based classification models have demonstrated remarkable success in medical imaging, particularly in diagnosing chest diseases using chest X-rays (CXR). Researchers have explored a variety of techniques for pneumonia classification, utilizing custom deep learning models, transfer learning, and ensemble-based feature extraction. Pretrained architectures have been widely adopted, with studies using AlexNet to classify CXR images into multiple disease categories, demonstrating the effectiveness of transfer learning in feature extraction [19]. Similarly, VGG19, DenseNet121, Xception, and ResNet50 have been employed in transfer learning-based approaches to enhance classification performance on the OCT-CXR dataset, emphasizing the importance of feature diversity in deep learning-based pneumonia detection [20]. Additionally, hybrid classification techniques combining VGG-16 and Xception with machine learning classifiers, such as support vector machines (SVM), k-nearest neighbors (KNN), random forest, and Naïve Bayes, have been investigated to improve feature discrimination and model robustness [21, 22]. While these methodologies demonstrate strong classification potential, they often suffer from increased computational costs, model complexity, or dataset dependency, limiting their generalizability across diverse clinical settings.

Despite significant advancements in deep learning for image classification, several challenges remain unresolved, particularly in medical imaging applications. Traditional CNN architectures such as VGG-16 and Inception offer strong feature extraction capabilities, but they often struggle with feature redundancy, inefficient spatial representation, and suboptimal classification accuracy in complex datasets. Many existing models lack generalizability across different datasets due to overfitting or dataset bias, limiting their applicability in real-world clinical settings. Additionally, most deep learning-based pneumonia classification approaches rely solely on CNNs without integrating feature selection techniques, leading to high-dimensional feature spaces that increase computational cost and reduce efficiency. Another critical limitation is the lack of interpretability in model decisions, which is essential for clinical adoption, as healthcare professionals require transparent and explainable AI models to support diagnostic decisions. Addressing these challenges necessitates an

approach that not only enhances feature extraction and selection but also improves generalization and interpretability in deep learning-based pneumonia detection.

To address these limitations, we propose B4-GraftingNet, a novel 87-layer convolutional neural network that integrates the strengths of VGG and Inception architectures. By combining hierarchical feature extraction with a diversified receptive field strategy, our model aims to enhance feature learning, improve generalizability, and provide interpretable results for medical imaging applications. The complete pipeline of this study is as follows:

- **Proposed B4-GraftingNet:** A novel 87-layer convolutional neural network model integrating VGG and Inception patterns to enhance feature extraction and spatial representation.
- **GraftingNet Technique:** Implemented a technique similar to Inception modules to propagate diverse receptive field information within the CNN network.
- **Training and Evaluation:** Trained the model on the OCT-CXR dataset, extracting features for both training and testing, while utilizing the NIH clinical dataset as a holdout evaluation set.
- **Data Augmentation:** Addressed dataset bias through techniques such as Gaussian noise and salt & pepper noise, while avoiding transformations like clipping, shifting, and flipping to maintain model robustness in challenging conditions.
- **Feature Optimization:** Optimized feature selection using Binary Particle Swarm Optimization (BPSO) to enhance classification performance.
- **Classifier Evaluation:** Evaluated various classifiers to determine the most effective approach for pneumonia detection in medical imaging applications.
- **Model Interpretability:** Integrated grad-CAM visualization to ensure model interpretability by highlighting key decision regions in CXR images.

The rest of this manuscript is structured as follows: Section 2 presents related studies, summarizing prior deep learning approaches for medical image classification. Section 3 describes the proposed methodology, detailing the B4-GraftingNet architecture, feature extraction process, BPSO-based feature selection, and classification methods. Experimental results and comparative evaluations are presented in Section 4, demonstrating the effectiveness of the proposed model. Section 5 discusses findings, limitations, and potential areas for improvement, while Section 6 concludes the study and outlines future research directions. Additional methodological details are provided in the supporting materials.

2 | Related Studies

2.1 | Studies Utilizing the OCT-CXR Dataset

The OCT-CXR dataset has been widely used for pneumonia classification, with several studies employing deep learning-based techniques to enhance diagnostic accuracy. Ibrahim et al. [19] used a pretrained AlexNet model to classify CXR images into four categories: COVID-19, non-COVID-19 viral pneumonia, bacterial pneumonia, and normal cases. Their approach achieved an accuracy of 91.43%, with a sensitivity of 94.0%, specificity of 96.0%, and precision of 92.0%. However, their study lacked feature optimization techniques, which could have further enhanced performance.

Salehi et al. [20] explored multiple pretrained models, including VGG19, DenseNet121, Xception, and ResNet50, for binary pneumonia classification. Their results demonstrated accuracies ranging from 83.0% to 86.8%, with sensitivities exceeding 91.0%, specificities between 78.0% and 86.0%, and F1-scores above 89.0%. While their study showed promising results, the absence of explainable AI techniques limited its interpretability for real-world clinical applications.

Sharma and Guleria [21] developed a hybrid deep learning approach by integrating VGG-16 with various classifiers such as support vector machines (SVM), k-nearest neighbors (KNN), random forest (RF), and naïve Bayes. Their model achieved an accuracy of 92.15%, with sensitivity, specificity, precision, and F1-score reaching 93.08%, 97.40%, 94.28%, and 93.70%, respectively. However, the study did not incorporate feature selection methods, which could have reduced computational overhead while maintaining high performance. The overview of related studies for the OCT-CXR dataset is summarized in Table 1.

2.2 | Studies Utilizing the NIH-CXR Dataset

The NIH CXR dataset has been widely used as a benchmark for pneumonia and tuberculosis classification. Several deep learning

TABLE 1 | Summary of recent studies on the OCT-CXR dataset for pneumonia classification, including accuracy (Acc), sensitivity (Sen), specificity (Spec), precision (Prec), F1-score (F1), and explainable AI (XAI) techniques used for model interpretability.

Reference	Year	Methodology	Acc (%)	Sen (%)	Spec (%)	Prec (%)	F1 (%)	XAI
Ibrahim et al. [19]	2024	Pretrained AlexNet for Pneumonia Classification	91.43	94.0	96.0	92.0	Not specified	No
Salehi et al. [20]	2021	VGG19, DenseNet121, Xception, ResNet50	83.0–86.8	91.0+	78.0–86.0	87.0	89.0+	No
Sharma and Guleria [21]	2023	VGG-16 + SVM, KNN, RF, Naïve Bayes	92.15	93.08	97.40	94.28	93.70	Activation Maps

TABLE 2 | Overview of recent studies on the NIH CXR dataset, presenting various deep learning methods, performance metrics (Acc, Sen, Spec, Prec, F1), and the use of XAI for enhancing clinical explainability.

Reference	Year	Methodology	Acc (%)	Sen (%)	Spec (%)	Prec (%)	F1 (%)	XAI
Zaidi, et al. [23]	2022	Inception–Resnet V2	85.4	84.8	Not specified	84.7	84.7	No
Moryani, Sood and Chaudhary [24]	2023	CNN + ResNet + SMOTE + Conventional ML	93.2	92.8	94.1	91.3	92.1	No
Choudhry, et al. [25]	2025	CheX-Net Deep Learning Model	97.50	96.54	97.11	Not specified	96.17	No

approaches have been explored to enhance diagnostic accuracy, as summarized in Table 2.

Zaidi, et al. [23] developed a tuberculosis (TB) classification model based on Inception–ResNet V2, achieving an accuracy of 85.4%, with a sensitivity of 84.8% and a precision of 84.7%, leading to an F1-score of 84.7%. Although the model performed well, the study lacked comparative evaluations with other architectures and did not explore feature selection techniques to further enhance classification performance.

Moryani, Sood and Chaudhary [24] applied a hybrid approach combining CNN, ResNet, SMOTE (synthetic minority oversampling technique), and conventional machine learning algorithms for pneumonia and TB classification, achieving an accuracy of 93.2%, with 92.8% sensitivity, 94.1% specificity, 91.3% precision, and an F1-score of 92.1%. However, this study did not incorporate explainable AI (XAI) techniques, making it difficult to interpret how the model made predictions, which is crucial in clinical applications

Choudhry, et al. [25] proposed CheX-Net, a deep learning-based model for chest disease classification, achieving 97.50% accuracy, 96.54% sensitivity, 97.11% specificity, and an F1-score of 96.17%. Although CheX-Net demonstrated high classification performance, the study did not specify precision values and lacked an explainability component to help clinicians understand its decision-making process. Despite this, CheX-Net significantly outperformed previous models in terms of diagnostic accuracy for pneumonia and other thoracic diseases.

2.3 | Drawbacks of Recent Pneumonia Classification Methods

- Most models are trained on a single dataset (e.g., OCT-CXR or NIH CXR) and struggle to generalize when tested on independent datasets due to domain shifts.
- Pneumonia datasets are often imbalanced, with more normal cases than pneumonia cases, causing models to be biased toward the majority class and reducing sensitivity for rare pneumonia cases.
- Most deep learning models function as black-box classifiers, limiting clinical trust due to the absence of explainability techniques like grad-CAM or saliency map visualizations.
- The lack of feature optimization in most studies results in high-dimensional feature sets, leading to overfitting and increased computational costs without significant performance gains.

These challenges highlight the need for more generalizable, computationally efficient, and interpretable deep learning models for pneumonia classification, which we address in our proposed B4-GraftingNet approach.

3 | Materials and Methods

This section details the proposed approach, which encompasses several major steps. The dataset and novel CNN architecture are discussed in the upcoming subsections. Dataset preparation and augmentation, feature extraction using the proposed pre-trained CNN model, feature selection using the binary particle swarm optimization (BPSO) algorithm, and classification are elaborated in the supporting manuscript (Section 2).

Figure 1 provides an overview of the proposed approach's architecture.

3.1 | Datasets

This research utilizes two standard chest X-ray datasets: OCT-CXR [26] and NIH Chest X-ray [27], both publicly available on Kaggle. The OCT-CXR dataset comprises images from pediatric patients at the China Medical Centre, categorized into normal and pneumonia cases. This dataset is divided into 2,682 images for training and validation and 624 images for testing, with each grayscale chest X-ray image measuring 64 × 64 pixels. Details of the images before and after data augmentation are provided in Table 3.

The NIH Chest X-ray dataset contains 108,948 X-ray images representing eight different chest illnesses, including approximately 1430 pneumonia images. Each grayscale image has a resolution of 1024 × 1024 pixels. In this study, we used the NIH Chest X-ray dataset solely as a holdout set.

3.2 | Dataset Preparation

The dataset consists of 1341 grayscale images for training and validation and 624 images for testing, categorized into two classes:

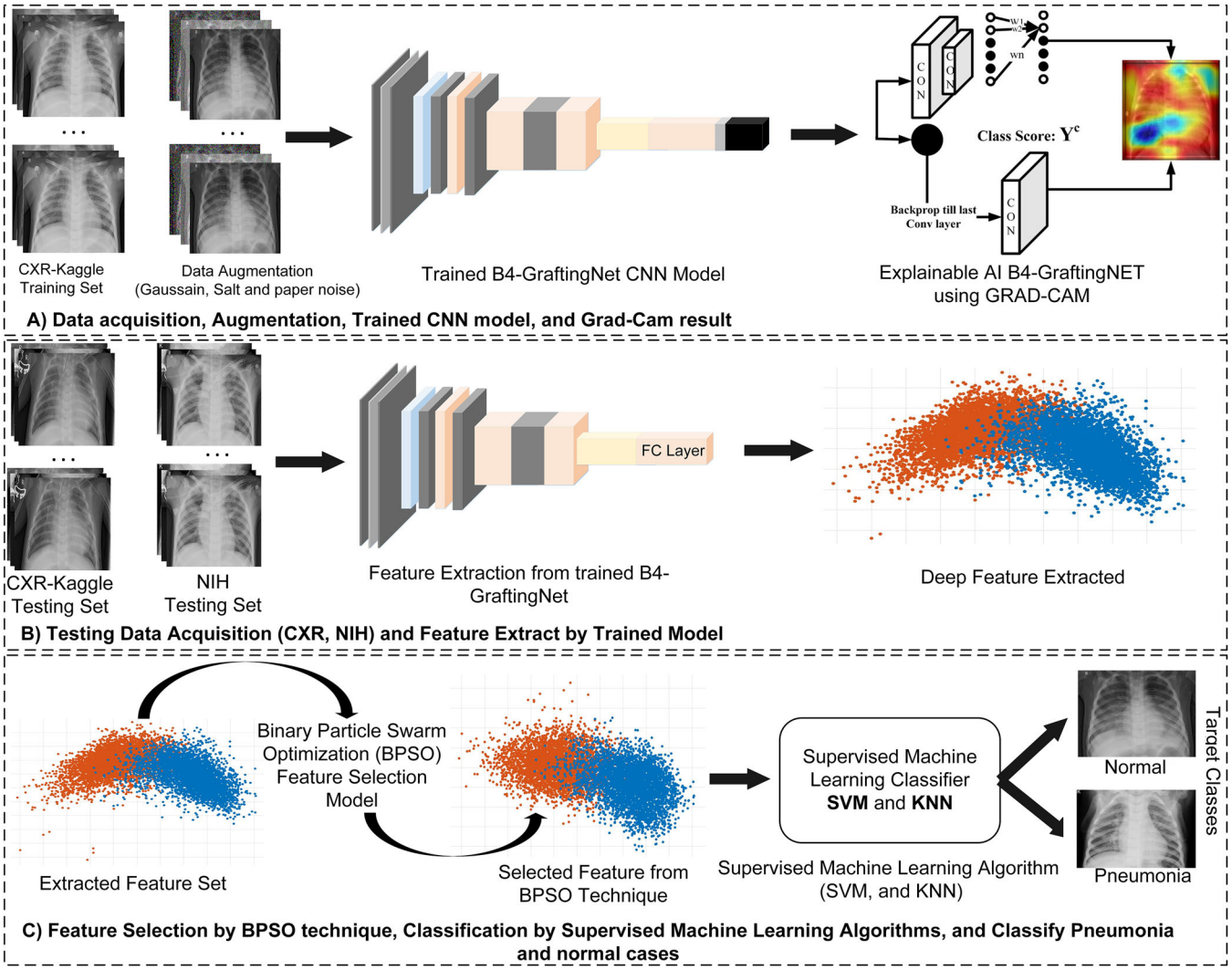

FIGURE 1 | Proposed methodology of B4-GraftingNet CNN for pneumonia classification where data augmentation, binary particle swarm optimization (BPSO), features extraction, and classification are mentioned in the supporting file.

TABLE 3 | Detail of experimental datasets before and after augmentation.

	OCT-CXR Dataset	
Class	Original	Augmented (including original)
Normal	1341	4023
Pneumonia	1341	4023
Total	**2682**	**8046**

The bold formatting displays the total number of samples, highlighting the overall dataset size before and after augmentation.

normal and pneumonia chest infections, within the OCT-CXR dataset. We identified 1341 healthy images in the OCT-CXR dataset. To ensure balance in every training and validation split, we equalized the number of images according to the lesser class. For the NIH Clinical dataset, we considered 1431 images of each class equally as a holdout set.

To enhance performance during training, a deep CNN model requires a larger dataset. Consequently, data augmentation techniques were employed to expand the training dataset. Two types of augmentation were used: Gaussian noise and salt-and-pepper noise (visualized in Figure 2), which increased the dataset size up to three times, including the original images. Detailed information about the dataset before and after augmentation is provided in Table 3.

Regarding augmentation techniques, we deliberately excluded clipping, shifting, flipping, tilting, and scaling to assess the model's performance in challenging conditions without artificially enhancing generalization. Instead, we focused on noise-based augmentation (Gaussian and salt-and-pepper noise) to increase dataset complexity, ensuring that the model learns robust feature representations even the dataset is complex. This choice allows us to evaluate how well the model adapts to real-world variations where imaging noise is a common factor. Additionally, as demonstrated in the experimental results (Section 4.1), particularly in the training phase of our proposed CNN model, these two augmentation techniques were sufficient to achieve strong performance, validating the effectiveness of our approach.

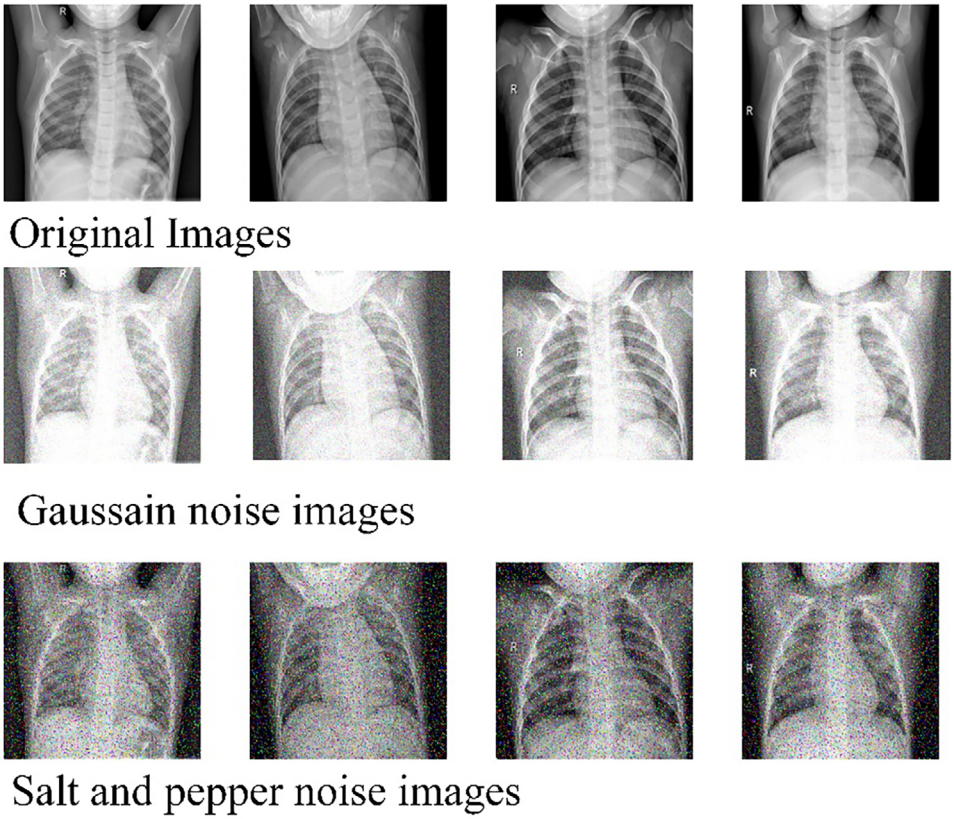

FIGURE 2 | OCT-CXR dataset: additive of noise to CXR images.

3.3 | Proposed CNN Based B4-Graftingnet Architecture

In the proposed architecture the *GB1* layer consists of ten layers distributed across three branches, each receiving input from the cross-channel normalization (*C1*) layer. The first branch comprises a convolutional (*C2*) layer. The second branch consists of three levels, incorporating a convolutional (*C3*) layer, Leaky ReLU (*LR1*), and another convolutional (*C4*) layer. The third branch includes a convolutional (*C5*) layer, Leaky ReLU (*LR2*), another convolutional (*C6*) layer, Leaky ReLU (*LR3*), and a final convolutional (*C7*) layer. These three branches are merged through an addition (*A1*) layer.

Following the *GB1* layer, sequential layers are arranged as batch normalization (*BN1*), *C8*, *BN2*, pooling (*P1*), *C9*, and cross-channel normalization (*CN2*) layers. The second cross-channel normalization (*CN2*) layer serves as the input for the subsequent grafting block (*GB2*), which mirrors the structure of *GB1*. *GB2* layers are combined through an addition (*A2*) layer. The output from *A2* proceeds to the next sequence, comprising *BN3*, *C16*, *CN3*, ReLU (*R1*), *P2*, *C17*, *BN4*, *C18*, and cross-channel normalization (*CN4*) layers.

After *CN4*, the third grafting block (*GB3*) is introduced, resembling *GB1* and *GB2*. Layers within *GB3* are concatenated using the addition (*A3*) layer. Following *GB3*, the layer sequence includes *BN5, C25, CN5, P3, C26, BN6, C27, R2*, and *CN6* layers. After *CN6*, the fourth and final grafting block (*GB4*) follows a structure similar to *GB1, GB2*, and *GB3*. *GB4* layers are unified through an addition (*A4*) layer, producing a single input for the subsequent batch normalization (*BN7*) layer.

Following the *GB4* layer, twenty layers are connected in sequence: *BN7, C34, R3, P4, C35, R4, C36, R5, C37, R6, P5*, fully connected (*FC1*), *R7*, dropout (*D1*), *FC2*, *R8*, *D2*, *FC3*, SoftMax (*S*), and output (*O*) layers. The proposed network includes three fully connected (*FC*) layers, two dropout (*D*) layers, one SoftMax (*S*) layer, and one output (*O*) layer. Table S1 presents the number of layers and their configuration details. A mathematical overview and output analysis at various levels of the proposed CNN network are provided. Further details on the overall architecture and convolutional neural network (CNN) configuration are available in [28].

3.4 | Proposed CNN Based B4-GraftingNet Model Development

The main contribution of this work is the proposed design of a CNN-based deep model named B4-GraftNet. The B4-GraftNet model was developed after studying CNN-based architectures, particularly VGG-16 [29]. VGG-16 consists of thirteen convolutional layers, three fully connected layers, fifteen ReLU layers, five max-pooling layers, two dropout layers, and a SoftMax layer. In total, it comprises 87 layers, including the input and output layers. Building on VGG-16, several additional layers were incorporated into the B4-GraftNet architecture, including batch normalization, leaky ReLU, and a branching mechanism. The architectural structure of B4-GraftNet is depicted in Figure 3.

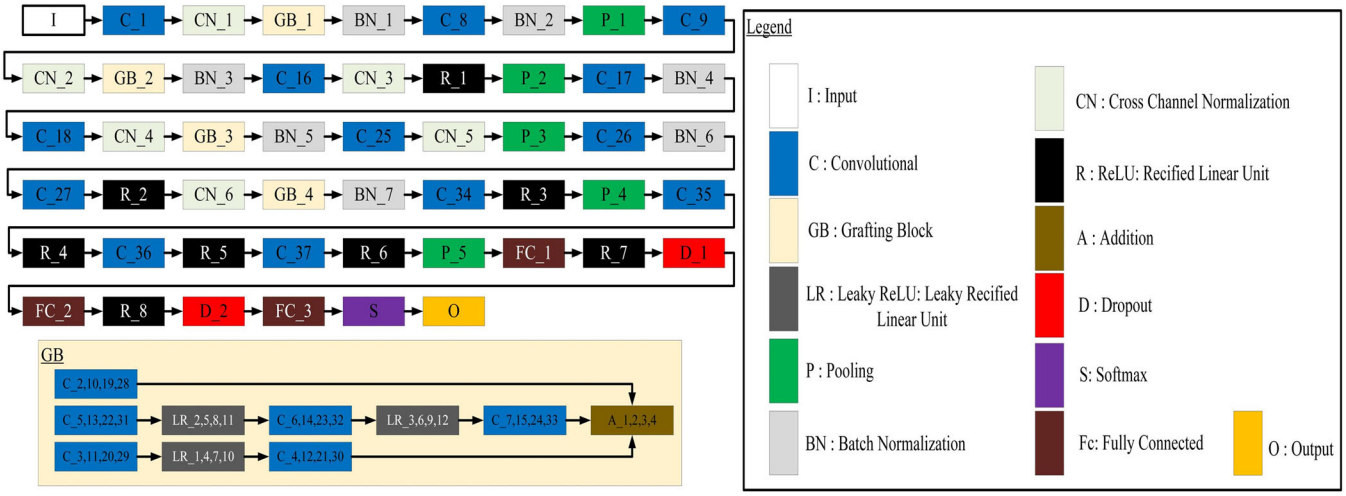

FIGURE 3 | Architectural diagram of the proposed B4-GraftingNet model. The network consists of convolutional layers (C1, C2, …), cross-channel normalization layers (CN), batch normalization layers (BN), grafting blocks (GB), activation functions (ReLU, Leaky ReLU), pooling layers (P), fully connected layers (FC), dropout layers (D), and a SoftMax classifier (S). The legend provides color-coded layer descriptions. The inset illustrates the structure of a grafting block (GB). For a detailed explanation of each layer and its functionality, refer to Section 3.3.

The B4-GraftNet architecture is designed with 87 layers to ensure an optimal balance between feature extraction capability, generalization, and computational efficiency. The choice of 87 layers is not arbitrary but is based on extensive empirical testing and theoretical considerations. Unlike shallower models, which struggle to extract hierarchical spatial patterns, deeper architectures such as B4-GraftNet can capture both low- and high-level representations, thereby enhancing classification performance. The integration of grafting blocks (GBs), batch normalization (BN), and cross-channel normalization (CN) ensures stable training and mitigates vanishing gradients, even at increased depth.

The model accepts images with dimensions of $227 \times 227 \times 3$. The proposed network begins with a convolutional layer, C_1, followed by a grafted branch (GB) resembling the fire module of SqueezeNet. The convolutional layer processes an input image or feature map $\mathcal{I}_{(x-1)}$ with ψ_x channels, where x represents the number of layers in the convolutional stack [30]. The output of the layer contains $\tilde{\psi}_x$ channels, computed using Equation (1):

$$\mathcal{I}_{\tilde{\psi},x} = \Omega_x \left(\sum_{\kappa} \Phi_{(x,\tilde{\psi},\kappa)} \circledast \mathcal{I}_{(\kappa,x-1)} + \mathbb{B}_{\tilde{\psi},x} \right) \tag{1}$$

where $\circledast$ represents the convolution operation, Φ is a filter with a depth of $\tilde{\psi}_x$ channels, $\mathbb{B}$ is the bias term, and Ω denotes the nonlinear activation function applied to each neuron in the input.

The proposed network employs two different activation functions: ReLU and Leaky ReLU [31]. The ReLU layer converts all negative values to zero, as shown in Equation (2).

$$\mathcal{I}_{(u,v)} = \max\left(0, \mathcal{I}_{(u,v)}\right) \tag{2}$$

where u and v represent row and column indices of the image matrix $\mathcal{I}$, respectively. In contrast, Leaky ReLU introduces a small slope instead of setting values strictly to zero. Leaky ReLU can be represented as in Equation (3):

$$\mathcal{Y} = \begin{cases} x, & x \geq 0 \\ 0.01x, & x < 0 \end{cases} \tag{3}$$

The proposed network integrates two distinct normalization layers: batch normalization (BN) and cross-channel normalization (CCN). For smaller batches, BN standardizes the inputs within a layer by computing the mean and variance for each feature. The batch mean is determined using Equation (4):

$$\mathbb{E}_{\text{Batch}} = \frac{1}{\zeta} \sum_{x=1}^{\zeta} \mathcal{F}_x \tag{4}$$

where Batch $= \{\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \ldots, \mathcal{F}_\zeta\}$ represents the feature set, and $\mathcal{F}$ denotes an original feature.

The variance of a small batch is computed in Equation (5) as follows:

$$\mathbb{V}_{\text{Batch}} = \frac{1}{\zeta} \sum_{j=1}^{\zeta} \left(\mathcal{F}_j - \mathbb{E}_{\text{Batch}}\right)^2 \tag{5}$$

Afterward, the features are normalized using Equation (6):

$$\mathcal{G}_y = \frac{\mathcal{F}_y - \mathbb{E}_{\text{Batch}}}{\sqrt{\mathbb{V}_{\text{Batch}} + \varepsilon}} \tag{6}$$

where ε is a small constant used to maintain numerical stability.

CCN aids in network generalization [32]. The cross-channel is represented by neighboring features in the CCN layer. CCN is mathematically expressed in Equation (7) as:

$$f = \frac{\mathcal{G}}{\left(\Theta + \frac{\alpha \cdot \beta}{\varpi}\right)^{\gamma}} \tag{7}$$

TABLE 4 | Training parameter of B4-GraftingNet.

Input size	Learning rate	Max epochs	Mini batch Sizes	Moment	Optimization
227 × 227 × 3	0.01	30	20	0.9	SGD

TABLE 5 | Number of extracted features using CNN models.

Deep CNN model	Number of extracted features	
	OCT-CXR dataset	NIH clinical dataset
B4-GraftingNet	624 × 4096	640 × 4096

where $\mathcal{G}$ represents the normalized feature before CCN, and f denotes the final transformed feature after CCN. Several hyperparameters Θ, α, and γ are used to determine the normalization value. β represents the sum of squared values, while ϖ corresponds to the channel's dimension.

3.5 | Training of B4-GraftingNet and Features Extraction

The proposed network, B4-GraftingNet, is trained using the OCT-CXR dataset [33]. The OCT-CXR dataset consists of two classes: normal and pneumonia images, and it is pre-divided into testing and training folders. Each class in the training set contains 1341 images, while the testing folder contains 624 images. To enhance the dataset for training, data augmentation techniques are applied. The training parameters of the proposed B4-GraftingNet are presented in Table 4, with further details of the trained model provided in the results section.

After successful training, features are extracted using the pre-trained B4-GraftingNet model on the OCT-CXR and NIH clinical testing and holdout datasets. The features extracted by B4-GraftingNet originate from the fully connected layer, *fc_2*, which generates a 1 × 4096-dimensional feature vector per image. All extracted features from the dataset are stored in a single feature matrix. Table 5 presents the total number of deep features extracted from both standard pneumonia datasets.

3.6 | BPSO for Features Optimization

In the feature extraction process, it is possible that some features contain ambiguous information, which can increase computational time and reduce classification accuracy. Therefore, the feature selection process is employed to extract the most relevant features from the feature set.

In this study, optimized feature selection is achieved through a nature-inspired algorithm known as Binary Particle Swarm Optimization (BPSO), proposed by Gaing [34]. BPSO is used to solve mathematical problems within a search space by efficiently determining the optimal path or solution to a given objective. This algorithm is inspired by natural behavior [35, 36] and, in general, requires fewer parameters while yielding highly accurate solutions. Moreover, PSO was designed to operate alongside common evolutionary methods such as the genetic algorithm (GA) [37]; however, it does not involve GA components such as selection, crossover, and mutations. As a search and optimization approach, the PSO algorithm produces efficient results [38].

In the search problem context, the initial population of PSO consists of particles associated with a swarm, generated randomly. Each particle in the population has two characteristics: position and velocity, represented by the position vector P_i and velocity vector V_i in Equations (8) and (9), which can be expressed as follows:

$$P_i = \{p_{i1}, p_{i2}, p_{i3}, \ldots\ldots, p_{in}\},\ \mathrm{i} = 1, 2, 3, \ldots, n, \tag{8}$$

where P_i represents the position vector of n swarm particles.

$$V_i = \{v_{i1}, v_{i2}, v_{i3}, \ldots\ldots, v_{in}\},\ \mathrm{i} = 1, 2, 3, \ldots, n \tag{9}$$

where V_i represents the velocity vector of m swarm particles. For finding the local best position, Equation (10) represents as follows:

$$Pbest_i = \{pb_{i1}, pb_{i2}, pb_{i3}, \ldots\ldots, pb_{in}\},\ \mathrm{i} = 1, 2, 3, \ldots,\ \mathrm{n}, \tag{10}$$

where $Pbest_i$ represents the optimal position of the i-th particle in a swarm of n particles. Each particle in the population is influenced by its velocity, personal best (*Pbest*) achieved at each iteration, and global best (*Gbest*), represented in Equation (11) as:

$$Gbest = \{gb_1, gb_2, gb_3, \ldots\ldots, gb_n\} \tag{11}$$

where, gb_n represents the global best prediction of 'n' swarm particles.

The particle movement across the search space is regulated by updating the position and velocity [39].The velocity is updated as follows in Equation (12):

$$v_{xy}^{t+1} = wv_{xy}^{t} + k_1 r_1 \left(p_{xy}^{t} - x_{xy}^{t}\right) + k_2 r_2 \left(g_y^{t} - x_{xy}^{t}\right) \tag{12}$$

and the position will be computed by Equation (13) as:

$$x_{xy}^{t+1} = x_{xy}^{t} + v_{xy}^{t+1} \tag{13}$$

The above-mentioned equations utilize 't' to represent iteration count, 'w' for inertia weight (ranging from 0 to 1), and *k1*and *k2* as cognitive and learning components. Random integers *r1* and *r2*, varying between 0 and 1, are also integrated. Each iteration involves updating particle positions and velocities, alongside computations for global and personal best values.

The position of each particle is represented in a binary string having a fixed value of 0 or 1 and velocity (V) represents the value

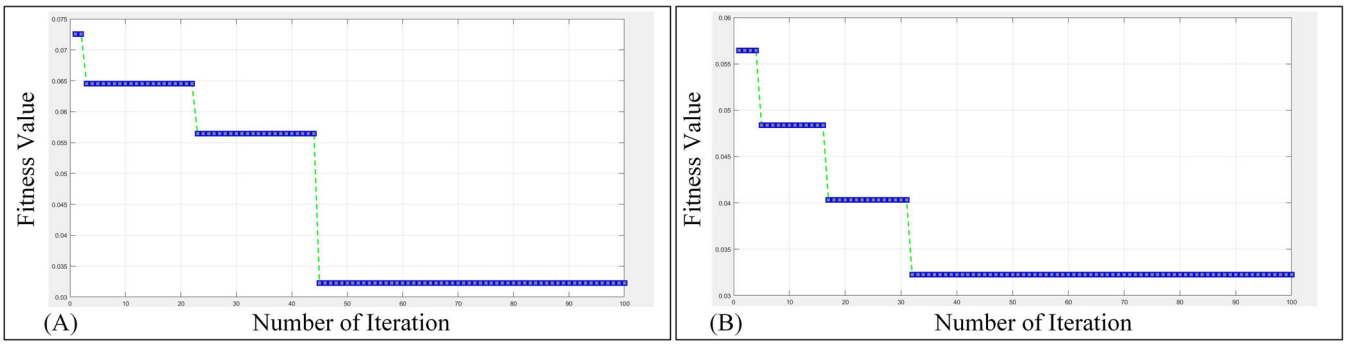

FIGURE 4 | Fitness value of every iteration for computing optimized features. (A) Fitness value on B4-GraftingNet feature set and (B) fitness value on fused feature set.

of the probability of particles with the fixed value of 1. The sigmoid activation function is used to translate the velocity value into a range between 0 and 1. The velocity and position of every particle are updated based on fitness function which is computed by KNN function. The value of particles updates according to the given Equations (14) and (15) as:

$$x_{ij}^{t+1} = \begin{cases} 1 & if \quad rand() < f\left(v_{ij}^{t+1}\right) \\ 0 & if \quad o.w \end{cases} \tag{14}$$

$$f\left(v_{ij}^{t+1}\right) = \frac{1}{1 + e^{-v_{ij}^{t+1}}} \tag{15}$$

where $rand()$ represents a random number from the [0,1] sequence distribution and x_{ij}^{t+1} is transformed into range 0 to 1 by a sigmoid function. Equations (12) and (15) normalize velocities into the range [0,1] and choose the set of features with the position set to 1 to produce the optimal answer from the binary vector of particle position. After the feature selection, the feature vector obtained has 624 × 2045 dimensions on OCT-CXR dataset and 1920 × 4071 dimensions on NIH clinical dataset, respectively. The fitness value of every iteration is shown in Figure 4. The PSO algorithm and its basic steps can be studied from [40, 41].

The PSO algorithm finds optimized features at 44th iteration by using B4-GraftingNet feature set and similarly by using fused feature vectors at 32nd iteration. The fitness value remains constant up to 100-th iteration.

3.7 | Classification

After the features selection process, an optimized features vector space is obtained for classification. The multiple supervised machine learning algorithms are used to classify normal and pneumonia images with efficient performance. The implemented classifiers are based on SVM [42] and KNN [43] variants. Furthermore, the variants of SVM include line SVM (L-SVM) [44], quadratic SVM (Q-SVM) [45], fine Gaussian SVM (FG-SVM) [46], medium Gaussian SVM (MG-SVM) [47], cubic SVM (C-SVM) [48], and coarse Gaussian SVM (CG-SVM) [49]. The decision function of SVM classifiers is computed by the number of support vectors, number of kernels function, and biased value expressed in Equation (16).

$$Y = \sum_{i=1}^{n} W_i K\left(v, v_i\right) + bv \tag{16}$$

In this equation, Y shows the decision output having value [−1,1], v represents support vectors, and kernel function is applied of every input vector represented by $K\ (v,\ vi)$. The biased factor is also added with the final weight of each kernel function [50]. SVM classifier and its kernel explanation can be studied from [51–53]. Other types of KNN classifiers used are cosine KNN (C-KNN) [54], coarse KNN (CR-KNN), and fine KNN (F-KNN). KNN classifiers calculate the similarity between the feature vectors based on Euclidian distance. The basic formula of the similarity measure in KNN algorithm is given in Equation (17).

$$s\ (a, b) = \sqrt{\sum_{i=1}^{n} \left(b_i - a_i\right)^2} \tag{17}$$

A detailed explanation of KNN and its other types can be studied from [55, 56]. These classifiers are evaluated on standard performance measure metrics. The experimental results and their details are presented in the result section.

4 | Results and Discussion

This section contains a thorough description and discussion of the experiment results. For the analysis and comparison of the proposed technique, results are collected on nine different classifiers including SVM and KNN classifiers. The section starts with the discussion of performance metrics as mentioned in supporting manuscript (Section 3). Then, the training process of the proposed model has been discussed in the upcoming sub section. After that, experimental environmental setups have been elaborated along summary of all the experiments. For experiments, we have provided only the best experiment in the main manuscript while all other experiments have been discussed in detail with all possible results in supporting manuscript (Section 3). Finally, the proposed model's features visualization along Grad-Cam and experimental results have been described and analyzed.

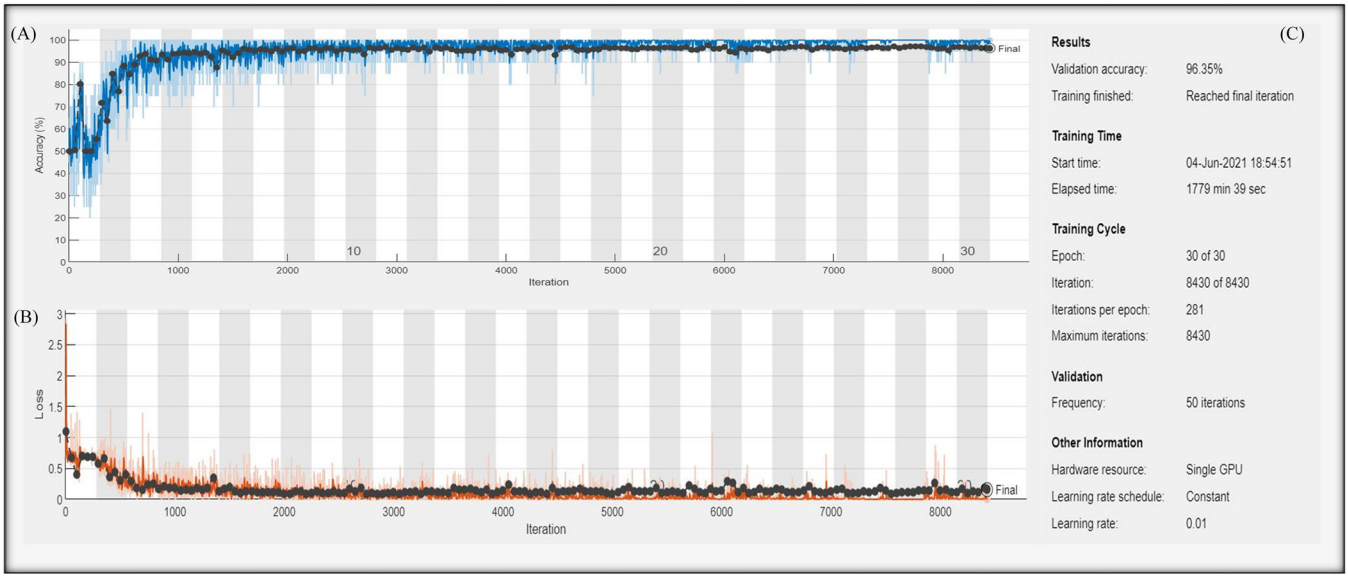

FIGURE 5 | Training performance of proposed CNN network.

4.1 | Training of Proposed CNN Network

The training process of B4-GraftingNet demonstrates the model's efficiency and stability in achieving high performance. The experiments were run on a Windows 11 PC with an 11th generation Intel Core i9 processor and an NVIDIA RTX 2060 GPU with 6 GB of RAM. MATLAB 2023b was chosen as the programming tool. In the training process, accuracy and loss were measured across 30 epochs, during which the required training accuracy was successfully achieved. The OCT-CXR dataset was divided into a 7:3 ratio for training and validation sets, resulting in a validation accuracy of 96.35%.

The training utilized optimal parameters, including stochastic gradient descent [57] for optimization, a mini-batch size of 20 to minimize the memory usage, 30 training iterations, and a learning rate of 0.01. Figure 5A–C visualizes the network's training accuracy, loss, and parameter values, illustrating the learning dynamics alongside validation accuracy. The training process was halted before overfitting occurred, as indicated by the opposing trends in validation loss and training loss. Following successful training, the last SoftMax layer was removed to retrieve feature vectors from the second fully connected (FC) layer of B4-GraftingNet. These results highlight the efficient training strategy, ensuring high performance while avoiding overfitting, and showcase the robustness of the B4-GraftingNet architecture for pneumonia classification.

After training, we performed different experiments based on features selection already described in Section 3.5. Here we have showed only the best achieved experiment while rest of them are mentioned in the supporting file under supporting results section.

4.2 | Experiments

A total of 11 experiments were conducted by selecting different numbers of optimal features to achieve efficient results. In all experiments, classification was performed using optimized features extracted from B4-GraftingNet. For the optimized feature set, the BPSO algorithm was applied to the fused feature matrix. All classifiers were tested on the deep selected features, and results in terms of various performance measures are presented in Table 6. Additionally, all performance measure equations, including accuracy, sensitivity, specificity, precision, F1-score, and others, are provided in the supporting document (Equations S1–S6), ensuring a comprehensive representation of all performance metrics.

The experimental results for the OCT-CXR 624-Images dataset (Experiments 1–8) demonstrate a clear trend of improved performance as the number of selected features increases. Initially, with 100 selected features, the model achieved an accuracy of 89.58%, but as the number of selected features increased, the performance metrics steadily improved. With 3000 selected features, the model obtained the highest accuracy (94.01%), sensitivity (94.22%), specificity (93.36%), precision (96.15%), and F1-score (95.18%), making it the best-performing configuration for the OCT-CXR dataset. Based on these results, we have chosen 3000 selected features as the optimal setting, as it consistently provided the highest classification performance.

For the NIH CXR 2862-Images dataset (Experiments 9–11), the results exhibited a different trend. The performance peaked with 1000 selected features, achieving an accuracy of 87.34% and balanced sensitivity (86.98%) and specificity (87.61%). However, increasing the number of selected features beyond 1000 did not yield significant improvements. At 1500 and 2000 selected features, the model exhibited some fluctuations in sensitivity and specificity but did not surpass the performance observed at 1000 selected features. Additionally, a test was conducted with 3000 selected features, but it resulted similar in accuracy and overall performance such as 1500 and 2000. Due to the similarity in results, the 3000-feature configuration for the NIH CXR dataset has not been included in the final analysis.

TABLE 6 | Summary of experiments performed for pneumonia classification.

Experiment no.	Dataset test	Selected features	Acc (%)	Sens (%)	Spec (%)	Pre (%)	F1-score (%)	Reference
1	OCT-CXR 624-Images	100	89.58	91.14	86.90	92.31	91.72	Table S2
2		250	90.6	91.00	88.39	93.33	92.15	Table S3
3		500	90.10	90.59	89.09	93.85	92.16	Table S4
4		750	91.10	91.48	88.89	93.59	92.52	Table S5
5		1000	90.22	91.44	88.11	93.08	92.25	Table S6
7		2000	90.71	91.92	88.60	93.33	92.62	Table S7
8		**3000**	**94.01**	**94.22**	**93.36**	**96.15**	**95.18**	Table 7
9	NIH CXR 2862-Images	**1000**	**87.34**	**86.98**	**87.61**	**87.71**	**87.34**	Table 8
10		1500	86.93	89.26	84.86	83.96	86.53	Table S8
11		2000	86.77	87.16	86.39	86.25	86.70	Table S9

The bold text highlights the best-performing experiments based on performance metrics for each dataset (OCT-CXR and NIH-CXR), indicating the optimal number of selected features in each case.

The accuracy drops from 94.01% (OCT-CXR) to 87.34% (NIH-CXR) is primarily due to dataset differences, domain shift, and evaluation conditions. The NIH dataset exhibits greater variability in pneumonia cases and image resolutions, making classification more challenging. Additionally, as the NIH dataset was used solely as a holdout test set rather than during model training, it provides a more realistic measure of generalization. Domain shift further contributes to this gap, as OCT-CXR primarily consists of pediatric pneumonia cases, whereas NIH-CXR includes a broader demographic, leading to variations in feature distribution. Moreover, performance trends within the NIH dataset show that increasing the number of selected features beyond 1000 does not necessarily improve classification results, highlighting the importance of dataset-specific feature selection.

To mitigate these challenges, domain adaptation techniques can enhance feature alignment between source and target datasets. Methods such as adversarial domain adaptation (e.g., domain-adversarial neural networks (DANN) [58]), feature transformation techniques (e.g., correlation alignment (CORAL) [59]), and self-supervised learning [60] could improve model generalization across datasets. These aspects will be further elaborated in the discussion section, and future work will explore domain adaptation strategies to enhance cross-dataset robustness.

It is important to note that our selection of the optimal feature range was based on test results rather than validation results. While validation results provide an indication of how well the model generalizes, test results are the true measure of the model's real-world effectiveness. By focusing on test results, we ensured that the selected feature configurations were optimized for actual deployment rather than just theoretical validation. Thus, for the OCT-CXR dataset, we identified 3000 selected features as the best-performing choice, while for the NIH CXR dataset, 1000 selected features provided the most stable and reliable results. The inclusion of 1500 and 2000 selected features for NIH CXR serves to illustrate performance variations, but our findings confirm that features selection from our proposed model is better with higher number of features.

4.3 | Experiment #8 (3000 Selected Features, OCT-CXR Dataset)

In this experiment, B4-GraftingNet deep model is used for features extraction. After this step, retrieved features are fed into BPSO algorithm for optimized features selection. This experiment is based on 3000 selected features from the optimized features set. The final size of optimal features matrix obtained becomes 624×3000 dimensions. All findings are based on SVM and KNN classifiers, which are supervised machine learning algorithms. In the testing phase, Q-SVM achieved the highest score of 94.01% accuracy. C-SVM is the second-best result in terms of accuracy, with a score of 93.75%. In KNN classifiers, best observation of 91.35% accuracy is grafted by W-KNN. This experiment has shown better results as compared to the other experiments with the observation that deep features from B4-GraftedNet at 3000 are optimized features and have obtained the highest result as compared to other observations of the experiment on OCT-CXR dataset having 624 images. In addition, Table 7 presents performance measures of classification results. Figure 6 depicts time slot and prediction speed of implemented classifiers on 3000 selected features space.

The confusion matrix and ROC curve of all classes of the experiment are shown in Figure 7 in left and right panels, respectively. The correctly predicted samples in each class have been shown diagonally on the left panel. This experiment has attained the best classification rate in all performance measures. The ROC curve and AUC score of 0.97 of this experiment has been shown in right panel.

4.4 | Experiment #9 (1000 Selected Features, NIH Clinical Augmented Dataset)

In Experiment #9, a total of 1000 features were selected from the BPSO-optimized feature set, and the dataset used for evaluation was the NIH clinical augmented dataset, as detailed in Section 3.1. The final experimental feature vector had dimensions of 2862×1000, ensuring a comprehensive representation of extracted

TABLE 7 | Performance of Experiment #8 (3000 features, OCT-CXR dataset).

Algorithm	Acc (%)	Sens (%)	Spec (%)	Pre (%)	FNR (%)	F1-score (%)
L-SVM	93.27	93.72	92.48	95.64	6.28	94.67
Q-SVM	**94.01**	**94.22**	**93.36**	**96.15**	**5.78**	**95.18**
C-SVM	93.75	93.98	93.33	96.15	6.02	95.06
MG-SVM	92.47	92.56	92.31	95.64	7.44	94.07
CG-SVM	89.90	88.11	93.85	96.92	11.89	92.31
F-KNN	89.26	91.09	86.15	91.79	8.91	91.44
M-KNN	90.54	89.50	92.68	96.15	10.50	92.71
Cos-KNN	91.19	90.95	91.63	95.38	9.05	93.12
W-KNN	91.35	90.98	92.06	95.64	9.02	93.25

Q-SVM is highlighted in bold for the OCT-CXR dataset as it achieved the highest performance metrics.

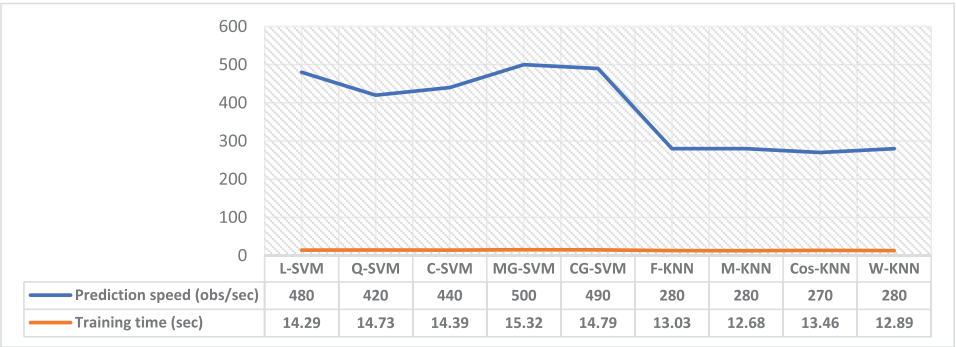

FIGURE 6 | Plot of prediction speed and training time of Experiment #8.

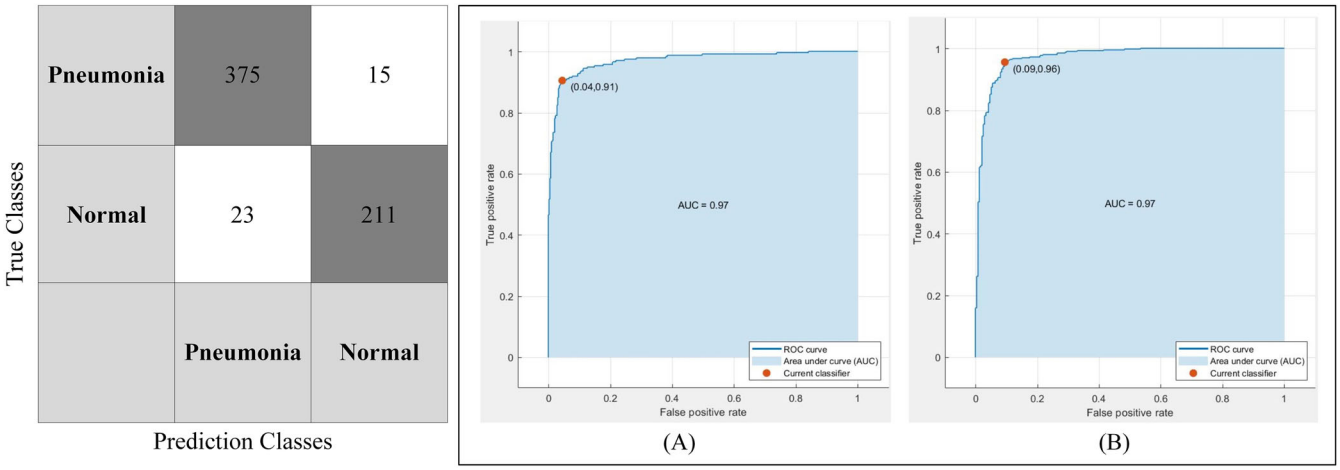

FIGURE 7 | Left image is confusion matrix of best outcomes (3000 features) of Experiment #8 on Q-SVM classifier; Right image is showing ROC and AUC of all classes with best outcomes (3000 features) of Experiment #8 on Q-SVM. (A) Normal; (B) pneumonia.

features. To minimize bias and enhance the reliability of classification, 5-fold cross-validation was performed on the feature matrix. The selected feature set was then classified using support vector machine (SVM) and K-nearest neighbors (KNN) classifiers to distinguish between pneumonia and normal cases. Among the classifiers, the Fuzzy KNN (F-KNN) algorithm demonstrated superior performance, achieving the highest classification accuracy of 87.34%, along with a sensitivity of 86.98%, specificity of 87.61%, and an F1-score of 87.34%. The Cubic SVM (C-SVM) followed as the second-best classifier, attaining an accuracy of 85.52%, sensitivity of 86.51%, specificity of 84.58%, and an F1-score of 85.32%. Additionally, quadratic SVM (Q-SVM) and weighted KNN (W-KNN) exhibited moderately high classification results, recording 81.77% and 79.64% accuracy, respectively. However, linear SVM (L-SVM), medium KNN (M-KNN), and cosine KNN (Cos-KNN) demonstrated relatively lower classification

TABLE 8 | Performance results of Experiment #9 (1000 features, NIH clinical augmented dataset).

Algorithm	Acc (%)	Sen (%)	Spec (%)	Pre (%)	FNR (%)	F1-score (%)
L-SVM	71.61	75.59	68.71	63.85	24.15	69.23
Q-SVM	81.77	83.52	80.20	79.17	16.48	81.28
C-SVM	85.52	86.51	84.58	84.17	13.49	85.32
MG-SVM	75.21	79.37	72.08	68.13	20.63	73.32
CG-SVM	62.92	68.51	59.92	47.81	31.49	56.32
F-KNN	**87.34**	**86.98**	**87.61**	**87.71**	**13.02**	**87.34**
M-KNN	69.32	68.72	69.97	70.94	31.28	69.81
Cos-KNN	67.76	67.03	68.55	69.90	32.97	68.43
W-KNN	79.64	82.97	76.92	74.58	17.03	78.55

F-KNN is highlighted in bold for the NIH dataset as it achieved the highest performance metrics.

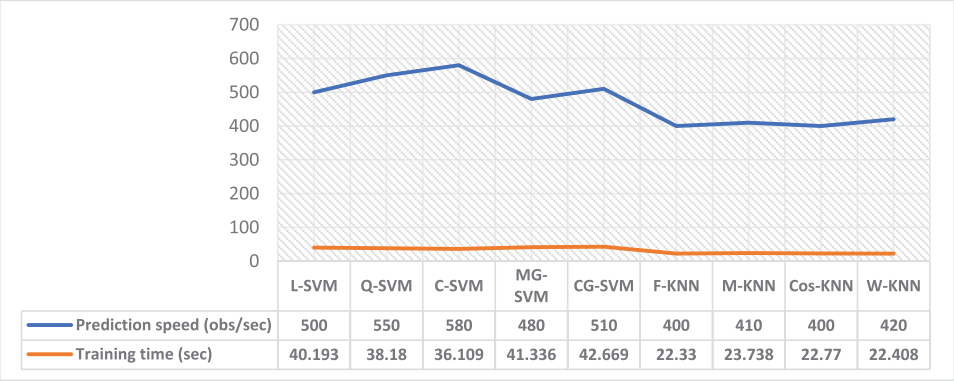

FIGURE 8 | Plot of prediction speed and training time of Experiment #9.

performance, with accuracy values ranging between 67.76% and 71.61%. The coarse Gaussian SVM (CG-SVM) performed the worst, achieving an accuracy of only 62.92%, with a significantly high false negative rate (FNR) of 31.49%, suggesting a considerable misclassification of pneumonia cases. The detailed performance metrics, including sensitivity, specificity, precision, and F1-score for each classifier, are summarized in Table 8, which provides a comparative analysis of the classification results. Furthermore, Figure 8 illustrates the training time and prediction speed of each classifier, offering insight into their computational efficiency and practical applicability for real-time diagnosis. The results indicate that F-KNN and C-SVM are the most suitable classifiers for pneumonia detection using the NIH clinical augmented dataset, outperforming other approaches in terms of both accuracy and robustness.

4.5 | Explainable-AI (XAI) Results of B4-Graftingnet Architecture by Using GRAD-CAM

4.5.1 | XAI on OCT-CXR

To understand the model's operation and its improved classification performance, we visualized the filters and features of the convolutional layers. This visualization reveals how the neural network processes information through various convolutional layers, as shown in Figure 9A–L. The visualizations demonstrate that the convolutional layers effectively preserve and refine the region of interest in the patient's chest, which forms the basis for accurate classification. Consequently, our model achieves better performance compared to previous techniques, as discussed in the following section. In addition, we have also utilized the last convolutional layer's weight mapping by gradient-weighted class activation mapping (grad-CAM) method to visualize the model attention pixels [61]. Grad-CAM creates heat maps in significant areas of the image. Grad-CAM takes an image as input and passes it to the SoftMax function of a pre-trained proposed model to predict the image's label. The predicted label, full trained model, and any convolutional layer (usually last) are used to compute grad-CAM results. Figure 9 shows that model activations on the input image and predicted class are greater in the last convolutional layer. The model's activations of normal image are almost equal in all regions. In cases of pneumonia, grad-CAM highlights pixels in the lungs. To predict the label, this research's model used these highly active areas in the input image.

4.5.2 | XAI on NIH-CXR

The grad-CAM (gradient-weighted class activation mapping) results presented for the NIH-CXR dataset highlight the explainability and interpretability of the B4-GraftingNet architecture, as illustrated in Figure 10. Grad-CAM is a powerful visualization technique that overlays a heatmap on input chest X-ray images

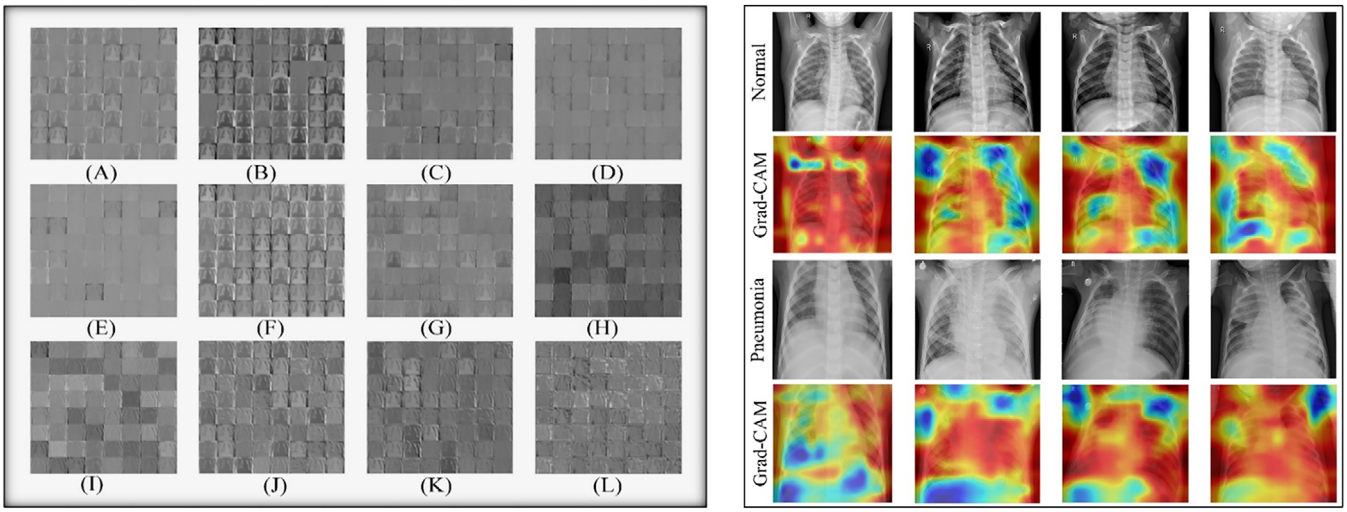

FIGURE 9 | Left side is showing activations of B4-GraftingNet against a query image at different convolutional layers (A) C_1 (B) C_2 (C) C_4 (D) C_6 (E) C_7 (F) C_10 (G) C_12 (H) C_23 (I) C_24 (J) C_26 (K) C_27 (L) C_37. Here, C refers to convolutional layers, where lower-indexed layers (e.g., C_1, C_2) capture low-level features such as edges and textures, while deeper layers (e.g., C_23, C_27, C_37) extract higher-level semantic features. A detailed description of the network architecture and layer configurations can be found in Section 3.3: Proposed CNN-Based B4-GraftingNet Architecture. The right side presents the grad-CAM visualization, highlighting the most critical regions contributing to the model's decision.

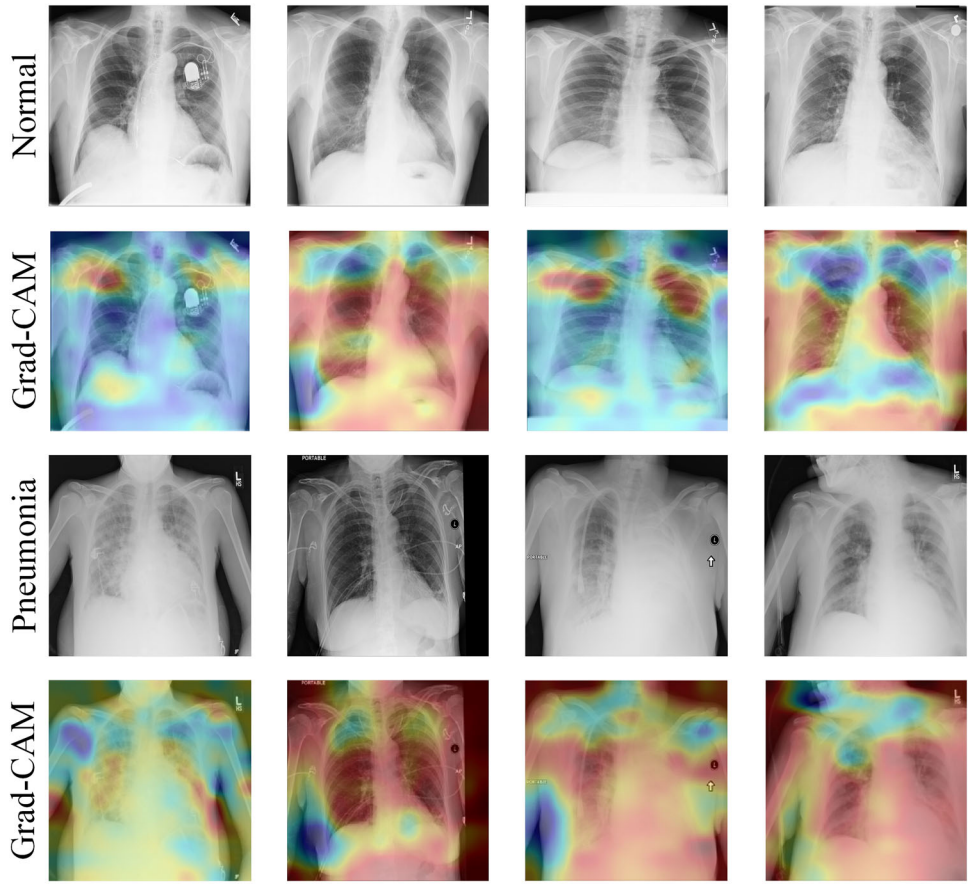

FIGURE 10 | Grad-CAM visualizations highlight the regions of interest in NIH-CXR images where the B4-GraftingNet model focuses for classifying normal and pneumonia cases.

TABLE 9 | Performance comparison of different methods for pneumonia classification using the OCT-CXR dataset. The proposed method demonstrates superior accuracy, sensitivity, specificity, precision, and F1-score, utilizing grad-CAM for enhanced explainability.

Reference	Year	Dataset	Acc (%)	Sens (%)	Spec (%)	Pre (%)	F1 (%)	XAI
Ibrahim et al. [19]	2024	Pretrained AlexNet for pneumonia classification	91.43	94.0	96.0	92.0	Not specified	No
Salehi et al. [20]	2021	VGG19, DenseNet121, Xception, ResNet50	83.0–86.8	91.0+	78.0–86.0	87.0	89.0+	No
Sharma and Guleria [21]	2023	VGG-16 + SVM, KNN, RF, naïve Bayes	92.15	93.08	97.40	94.28	93.70	Activation Maps
Proposed method	–	**OCT-CXR**	**94.01**	**94.22**	**93.36**	**96.15**	**95.18**	**Grad-CAM**

The bold text highlights the proposed methodology, emphasizing its performance in comparison to existing methods on the OCT-CXR dataset.

to indicate regions of interest that the model focuses on during classification. This ensures transparency in the decision-making process of the deep learning model.

For normal cases, the grad-CAM visualizations show minimal or no significant activation in the chest regions, indicating the model correctly identifies the absence of abnormalities. The heatmaps for normal cases demonstrate the model's ability to focus on uniform lung areas, showcasing its proficiency in identifying the structural integrity of healthy lungs.

In contrast, for pneumonia cases, the grad-CAM heatmaps reveal intense activations in localized areas, particularly over regions with opacities or abnormal patterns associated with pneumonia. These heatmaps indicate that the model effectively detects signs of inflammation, fluid build-up, or infection in the lungs, which are hallmark indicators of pneumonia. The concentration of activations aligns well with radiologists' observations, demonstrating the model's capability to mimic expert-level diagnostic focus.

The use of grad-CAM in this study not only validates the model's classification performance but also provides critical insights into its diagnostic reasoning. By visualizing how B4-GraftingNet identifies pneumonia-specific patterns, this method fosters trust and confidence among clinicians, making it a valuable tool for real-world applications. Furthermore, these visualizations highlight the robustness of the proposed model in interpreting complex imaging data from the NIH-CXR dataset, even under varying imaging conditions. While this study primarily focuses on the technical evaluation of grad-CAM heatmaps, future work will include validation by medical experts to assess their clinical interpretability and applicability.

The application of this technique proves valuable for both neonates and adults, proposing a useful approach to detect and classify lung diseases across different age groups. Additionally, in medical imaging, the utilization of X-ray, CNN, and explainable-AI tools has demonstrated efficacy in various problems such as cancers, heart disease, etc., by unveiling the hidden insights that will assist the researchers and doctors in their understandings for a particular problem. Also, it has capability to be utilized in different domains such as materials science for non-destructive testing and image recognition tasks in computer vision for more advancements.

5 | Comparison With Previous Studies

5.1 | Comparison on OCT-CXR Dataset

The proposed method demonstrates strong performance on the OCT-CXR dataset, achieving 94.01% accuracy, 94.22% sensitivity, 93.36% specificity, 96.15% precision, and an F1-score of 95.18% while incorporating grad-CAM for explainability, as described in Table 9. Compared to previous works, it shows a significant improvement in overall classification performance and interpretability.

Ibrahim et al. [19] utilized a pretrained AlexNet model for pneumonia classification, achieving 91.43% accuracy, 94.0% sensitivity, and 96.0% specificity. While their model performed well, it lacked feature optimization techniques and did not incorporate explainable AI, making it a black-box model with limited clinical interpretability.

Salehi et al. [20] explored multiple pretrained architectures such as VGG19, DenseNet121, Xception, and ResNet50, achieving accuracies ranging from 83.0% to 86.8% with sensitivities above 91.0%. However, their specificity was relatively lower (78.0–86.0%), and they did not incorporate explainability techniques, which limits their clinical usability. While these models demonstrated high sensitivity, the lack of interpretability and specificity trade-offs highlight the challenges of relying solely on pretrained networks.

Sharma and Guleria [21] applied VGG-16 with multiple classifiers such as SVM, KNN, random forest, and naïve Bayes, achieving 92.15% accuracy, 93.08% sensitivity, 97.40% specificity, and 94.28% precision, with an F1-score of 93.70%. They incorporated activation maps for visualization, which enhances model interpretability. However, their approach relied on multiple classifiers, which may increase computational complexity without significantly improving performance.

In contrast, the proposed B4-GraftingNet model outperforms or remains competitive with these prior methods by balancing high sensitivity, specificity, and precision, while also providing a robust explainability mechanism (grad-CAM). This makes it more suitable for real-world clinical applications, ensuring both performance and interpretability.

TABLE 10 | Performance comparison of different methods for pneumonia classification using the NIH-CXR dataset. The proposed method demonstrates superior accuracy, sensitivity, specificity, precision, and F1-score, utilizing grad-CAM for enhanced explainability.

Reference	Year	Methodology	Acc (%)	Sen (%)	Spec (%)	Prec (%)	F1 (%)	XAI
Zaidi, et al. [23]	2022	Inception–Resnet V2	85.4	84.8	Not specified	84.7	84.7	No
Moryani, Sood and Chaudhary [24]	2023	CNN + ResNet + SMOTE + Conventional ML	93.2	92.8	94.1	91.3	92.1	No
Choudhry, et al. [25]	2025	CheX-Net deep learning model	97.50	96.54	97.11	Not specified	96.17	Mask generation
Proposed method	–	**OCT-CXR**	**87.34**	**86.98**	**87.61**	**87.71**	**87.34**	Grad-CAM

The bold text highlights the proposed methodology, emphasizing its performance in comparison to existing methods on the NIH-CXR dataset.

5.2 | Comparison on NIH-CXR Dataset

The proposed method achieves 87.34% accuracy, 86.98% sensitivity, 87.61% specificity, 87.71% precision, and an F1-score of 87.34% on the NIH-CXR dataset as a holdout test set, demonstrating its generalization capability without prior exposure to the dataset during training, as detailed in Table 10. This is a notable distinction from previous studies, which used the NIH dataset for both training and validation, whereas our model was tested on it as an independent external evaluation. Despite this challenge, the proposed method remains comparable to state-of-the-art approaches.

Zaidi, et al. (2022) employed the Inception-ResNet V2 architecture for the classification of chest X-ray images, achieving an accuracy of 85.4%, which reflects moderate performance. The model demonstrated balanced sensitivity (84.8%) and precision (84.7%), indicating its ability to detect true positive cases and maintain precision in its predictions. However, the study did not provide details about specificity, which is a critical metric in medical imaging as it ensures the model's ability to correctly identify true negatives. Furthermore, the absence of explainability features, such as visualizations or interpretable outputs, limits its practical application in clinical settings where understanding the reasoning behind predictions is crucial for clinician trust and adoption.

Moryani, Sood, and Chaudhary (2023) utilized a hybrid CNN + ResNet model with SMOTE and conventional machine learning techniques, achieving 93.2% accuracy, 92.8% sensitivity, 94.1% specificity, and an F1-score of 92.1%. While their model achieved higher overall accuracy, it was trained, validated, and tested on NIH-CXR, benefiting from exposure to the dataset throughout development. The proposed method, despite being evaluated solely on a holdout set, achieved competitive performance, proving its robust generalization capabilities.

Choudhry and Iqbal (2025) introduced CheX-Net, a deep learning model designed for chest disease classification, achieving 97.50% accuracy, 96.54% sensitivity, 97.11% specificity, and an F1-score of 96.17%. Although this model outperformed others in terms of raw performance, it lacked explainability (XAI), making it less interpretable in clinical applications. Furthermore, CheX-Net was trained and validated using the NIH dataset, giving it an advantage over our model, which was strictly evaluated as a holdout. The inclusion of mask generation techniques added a level of interpretability, though limited to specific visualization contexts.

Despite these differences, the proposed B4-GraftingNet model delivers comparable results on an unseen dataset, proving its strong generalization capability. Additionally, grad-CAM explainability makes it a more interpretable model for real-world deployment. This highlights the effectiveness of our approach in learning robust feature representations that extend beyond training data, ensuring better reliability in diverse clinical settings.

6 | Limitations

Despite its strong performance, B4-GraftingNet has certain limitations. The training process is time-consuming, even on a high-end GPU (NVIDIA RTX 2060) and an Intel Core i9 processor. Additionally, while Gaussian noise and salt & pepper noise were used for data augmentation, incorporating a broader range of techniques such as clipping, rotation, and flipping may further enhance model robustness. Lastly, domain adaptation techniques could improve performance when applied to datasets with significant variations in imaging conditions.

7 | Conclusion

This study presents B4-GraftingNet, an advanced deep learning model that integrates the hierarchical feature extraction of VGG-16 with the diversified receptive field strategy of Inception, enhancing both classification accuracy and interpretability. Evaluated on pneumonia detection as a case study, the model achieves 94.01% accuracy on the OCT-CXR dataset and 87.34% accuracy on the NIH-CXR dataset, demonstrating its robust generalization capability, even when tested on an unseen dataset. The integration of feature selection using Binary Particle Swarm Optimization (BPSO) ensures that only the most relevant features are utilized, while grad-CAM explainability provides critical insights into the model's decision-making process. These enhancements make B4-GraftingNet highly effective for both medical and non-medical classification tasks. Compared to prior methods, the proposed model strikes a better balance between performance, computational efficiency, and explainability, making it a promising choice for real-world deployment.

Future studies will focus on optimizing the model for low-resource environments by reducing computational overhead and improving real-time deployment feasibility. Additionally, domain adaptation techniques will be explored to enhance cross-dataset generalization, ensuring broader applicability across diverse imaging conditions. Furthermore, clinical validation will be conducted to assess the model's interpretability and reliability, addressing all critical aspects for real-world medical implementation.

Author Contributions

Syed Adil Hussain Shah: conceptualization, data curation, formal analysis, methodology, resources, software, validation, visualization, writing – original draft, writing – review & editing. **Syed Taimoor Hussain Shah:** conceptualization, data curation, formal analysis, methodology, resources, software, validation, visualization, writing – original draft, writing – review & editing. **Abdul Muiz Fayyaz:** data curation, methodology, software, writing – original draft. **Syed Baqir Hussain Shah:** data curation, methodology, software, writing – original draft. **Mussarat Yasmin:** formal analysis, investigation, methodology, resources, validation, supervision, writing – review & editing. **Mudassar Raza:** formal analysis, investigation, methodology, resources, validation, visualization, supervision, writing – review & editing. **Angelo Di Terlizzi:** formal analysis, investigation, methodology, resources, visualization, supervision, writing – review & editing. **Marco Agostino Deriu:** formal analysis, funding acquisition, investigation, methodology, resources, validation, visualization, supervision, writing – review & editing.

Acknowledgements

Open access publishing facilitated by Politecnico di Torino, as part of the Wiley - CRUI-CARE agreement.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The data sets utilized in this study are both publicly accessible and cited in this publication under the datasets section.

References

1. N. Thakur, E. Bhattacharjee, R. Jain, B. Acharya, and Y.-C. Hu, "Deep Learning-Based Parking Occupancy Detection Framework Using Resnet and Vgg-16," *Multimedia Tools and Applications* 83, no. 1 (2024): 1941–1964.

2. R. Kaur and S. Singh, "A Comprehensive Review of Object Detection With Deep Learning," *Digital Signal Processing* 132 (2023): 103812.

3. P. Ma, C. Li, M. M. Rahaman, et al., "A State-of-the-Art Survey of Object Detection Techniques in Microorganism Image Analysis: From Classical Methods to Deep Learning Approaches," *Artificial Intelligence Review* 56, no. 2 (2023): 1627–1698.

4. S. F. Ahmed, M. S. B. Alam, M. Hassan, et al., "Deep Learning Modelling Techniques: Current Progress, Applications, Advantages, and Challenges," *Artificial Intelligence Review* 56, no. 11 (2023): 13521–13617.

5. K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv preprint, arXiv:1409.1556 (2014).

6. C. Szegedy, W. Liu, Y. Jia, et al, "Going Deeper With Convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2015).

7. K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2016).

8. A. Vaswani, N. Shazeer, N. Parmar, et al., "Attention Is All You Need," in *Advances in Neural Information Processing Systems (Curran Associates, Inc.)*, (2017).

9. F. Chollet, "Xception: Deep Learning With Depthwise Separable Convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2017).

10. X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: An Extremely Efficient Convolutional Neural Network for Mobile Devices," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2018).

11. S. T. H. Shah, S. A. H. Shah, I. I. Khan, et al., "Data-Driven Classification and Explainable-Ai in the Field of Lung Imaging," *Frontiers in Big Data* 7 (2024): 1393758.

12. S. B. Zaman, N. Hossain, M. T. U. S. Talha, K. Hasan, R. B. Zaman, and R. Khan, "Assessing the Risk of Antibiotic Resistance in Childhood Pneumonia: A Hospital-Based Study in Bangladesh," *Healthcare* 13, no. 3 (2025): 207.

13. D. Yao, Z. Xu, Y. Lin, and Y. Zhan, "Accurate and Intelligent Diagnosis of Pediatric Pneumonia Using X-Ray Images and Blood Testing Data," *Frontiers in Bioengineering and Biotechnology* 11 (2023): 1058888.

14. M. Palmer, J. A. Seddon, M. M. van der Zalm, et al., "Optimising Computer Aided Detection to Identify Intra-Thoracic Tuberculosis on Chest X-Ray in South African Children," *PLOS Global Public Health* 3, no. 5 (2023): e0001799.

15. D. M. Le Roux and H. J. Zar, "Community-Acquired Pneumonia in Children—A Changing Spectrum of Disease," *Pediatric Radiology* 47, no. 11 (2017): 1392–1398.

16. T. Nguyen, T. Tran, C. Roberts, S. Graham, and B. Marais, "Child Pneumonia–Focus on the Western Pacific Region," *Paediatric Respiratory Reviews* 21 (2017): 102–110.

17. K. Watkins and D. Sridhar, "Pneumonia: A Global Cause Without Champions," *Lancet* 392, no. 10149 (2018): 718–719.

18. P. Hamet and J. Tremblay, "Artificial Intelligence in Medicine," *Metabolism* 69 (2017): S36–S40.

19. A. U. Ibrahim, M. Ozsoz, S. Serte, F. Al-Turjman, and P. S. Yakoi, "Pneumonia Classification Using Deep Learning From Chest X-Ray Images during Covid-19," *Cognitive Computation* 16, no. 4 (2024): 1589–1601.

20. M. Salehi, R. Mohammadi, H. Ghaffari, N. Sadighi, and R. Reiazi, "Automated Detection of Pneumonia Cases Using Deep Transfer Learning With Paediatric Chest X-Ray Images," *The British Journal of Radiology* 94, no. 1121 (2021): 20201263.

21. S. Sharma and K. Guleria, "A Deep Learning Based Model for the Detection of Pneumonia From Chest X-Ray Images Using Vgg-16 and Neural Networks," *Procedia Computer Science* 218 (2023): 357–366.

22. S. A. H. Shah, S. T. H. Shah, A. Buccoliero, et al., "Explainable AI-Based Skin Cancer Detection Using Cnn, Particle Swarm Optimization and Machine Learning," *Journal of Imaging* 10, no. 12 (2024): 332.

23. S. Z. Y. Zaidi, M. U. Akram, A. Jameel, and N. S. Alghamdi, "A Deep Learning Approach for the Classification of Tb From Nih Cxr Dataset," *IET Image Processing* 16, no. 3 (2022): 787–796.

24. B. Moryani, K. Sood, and K. Chaudhary, "A Deep Learning Approach for the Classification of Tuberculosis and Pneumonia Using Nih Dataset," in *2023 International Symposium on Networks, Computers and Communications (ISNCC)* (IEEE, 2023).

25. I. A. Choudhry, S. Iqbal, M. Alhussein, A. N. Qureshi, K. Aurangzeb, and R. A. Naqvi, "Transforming Lung Disease Diagnosis With Transfer Learning Using Chest X-Ray Images on Cloud Computing," *Expert Systems* 42, no. 2 (2025): e13750.

26. D. Kermany, K. Zhang, and M. Goldbaum, "Labeled Optical Coherence Tomography (Oct) and Chest X-Ray Images for Classification," *Mendeley Data* 2, no. 2 (2018): 651.

27. R. Summers, *NIH Chest X-Ray Dataset of 14 Common Thorax Disease Categories* (NIH Clinical Center, 2019).

28. R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional Neural Networks: An Overview and Application in Radiology," *Insights Into Imaging* 9, no. 4 (2018): 611–629.

29. O. N. Belaid and M. Loudini, "Classification of Brain Tumor by Combination of Pre-Trained Vgg16 Cnn," *Journal of Information Technology Management* 12, no. 2 (2020): 13–25.

30. S. Balocco, M. González, R. Ñanculef, P. Radeva, and G. Thomas, "Calcified Plaque Detection in Ivus Sequences: Preliminary Results Using Convolutional Nets," in *International Workshop on Artificial Intelligence and Pattern Recognition* (Springer, 2018).

31. Y. Liu, X. Wang, L. Wang, and D. Liu, "A Modified Leaky Relu Scheme (MLRS) for Topology Optimization With Multiple Materials," *Applied Mathematics and Computation* 352 (2019): 188–204.

32. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet Classification With Deep Convolutional Neural Networks," *Advances in Neural Information Processing Systems* 25 (2012): 1097–1105.

33. D. Kermany, K. Zhang, and M. Goldbaum, "Labeled Optical Coherence Tomography (Oct) and Chest X-Ray Images for Classification," *Mendeley Data Version* 2 (2018), https://doi.org/10.17632/rscbjbr9sj.2.

34. D. Wang, D. Tan, and L. Liu, "Particle Swarm Optimization Algorithm: An Overview," *Soft Computing* 22 (2018): 387–408.

35. M. Hajihassani, D. J. Armaghani, and R. Kalatehjari, "Applications of Particle Swarm Optimization in Geotechnical Engineering: A Comprehensive Review," *Geotechnical and Geological Engineering* 36, no. 2 (2018): 705–722.

36. K.-H. Chen, K.-J. Wang, M.-L. Tsai, et al., "Gene Selection for Cancer Identification: A Decision Tree Model Empowered by Particle Swarm Optimization Algorithm," *BMC Bioinformatics* 15, no. 1 (2014): 1–10.

37. T. Li, G. Shao, W. Zuo, and S. Huang, "Genetic Algorithm for Building Optimization: State-of-the-Art Survey," in *Proceedings of the 9th International Conference on Machine Learning and Computing* (Association for Computing Machinery, 2017).

38. E. Naderi, M. Pourakbari-Kasmaei, and H. Abdi, "An Efficient Particle Swarm Optimization Algorithm to Solve Optimal Power Flow Problem Integrated With Facts Devices," *Applied Soft Computing* 80 (2019): 243–262.

39. J. Cervantes, F. Garcia-Lamont, L. Rodriguez, A. López, J. R. Castilla, and A. Trueba, "PSO-Based Method for SVM Classification on Skewed Data Sets," *Neurocomputing* 228 (2017): 187–197.

40. J. Kennedy and R. Eberhart, "Particle Swarm Optimization," in *Proceedings of ICNN'95-International Conference on Neural Networks* (IEEE, 1995).

41. J. C. Bansal, "Particle Swarm Optimization," in *Evolutionary and Swarm Intelligence Algorithms* (Springer, 2019).

42. W. S. Noble, "What Is a Support Vector Machine?," *Nature Biotechnology* 24, no. 12 (2006): 1565–1567.

43. L. Peterson, "K-Nearest Neighbor," *Scholarpedia* 4, no. 2 (2009): 1883.

44. Y.-W. Chang and C.-J. Lin, "Feature Ranking Using Linear SVM," in *Causation and Prediction Challenge* (PMLR, 2008).

45. I. Dagher, "Quadratic Kernel-Free Non-Linear Support Vector Machine," *Journal of Global Optimization* 41, no. 1 (2008): 15–30.

46. P. Virdi, Y. Narayan, P. Kumari, and L. Mathew, "Discrete Wavelet Packet Based Elbow Movement Classification Using Fine Gaussian SVM," in *2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)* (IEEE, 2016).

47. J.-Z. Feng, Y. Wang, J. Peng, M.-W. Sun, J. Zeng, and H. Jiang, "Comparison Between Logistic Regression and Machine Learning Algorithms on Survival Prediction of Traumatic Brain Injuries," *Journal of Critical Care* 54 (2019): 110–116.

48. U. Jain, K. Nathani, N. Ruban, A. N. J. Raj, Z. Zhuang, and V. G. Mahesh, "Cubic SVM Classifier Based Feature Extraction and Emotion Detection From Speech Signals," in *2018 International Conference on Sensor Networks and Signal Processing (SNSP)* (IEEE, 2018).

49. B. S. Bhati and C. S. Rai, "Intrusion Detection Technique Using Coarse Gaussian SVM," *International Journal of Grid and Utility Computing* 12, no. 1 (2021): 27.

50. Z. Soumaya, B. D. Taoufiq, N. Benayad, K. Yunus, and A. Abdelkrim, "The Detection of Parkinson Disease Using the Genetic Algorithm and SVM Classifier," *Applied Acoustics* 171 (2021): 107528.

51. S. Rüping, "SVM Kernels for Time Series Analysis," Technical Report (2001).

52. N.-E. Ayat, M. Cheriet, and C. Y. Suen, "Automatic Model Selection for the Optimization of Svm Kernels," *Pattern Recognition* 38, no. 10 (2005): 1733–1745.

53. B. Haasdonk, "Feature Space Interpretation of SVMs With Indefinite Kernels," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27, no. 4 (2005): 482–492.

54. E. Prasetyo, R. Purbaningtyas, and R. D. Adityo, "Cosine K-Nearest Neighbor in Milkfish Eye Classification," *International Journal of Intelligent Engineering and Systems* 13, no. 3 (2020): 11–25.

55. A. Lamba and D. Kumar, "Survey on KNN and Its Variants," *International Journal of Advanced Research in Computer and Communication Engineering* 5, no. 5 (2016): 430–435.

56. Y. Xu, Q. Zhu, Z. Fan, M. Qiu, Y. Chen, and H. Liu, "Coarse to Fine K Nearest Neighbor Classifier," *Pattern Recognition Letters* 34, no. 9 (2013): 980–986.

57. E. Yazan and M. F. Talu, "Comparison of the Stochastic Gradient Descent Based Optimization Techniques," in *2017 International Artificial Intelligence and Data Processing Symposium (IDAP)* (IEEE, 2017).

58. Y. Ganin, E. Ustinova, H. Ajakan, et al., "Domain-Adversarial Training of Neural Networks," *Journal of Machine Learning Research* 17, no. 59 (2016): 1–35.

59. X.-C. Zhong, Q. Wang, D. Liu, et al., "A Deep Domain Adaptation Framework With Correlation Alignment for Eeg-Based Motor Imagery Classification," *Computers in Biology and Medicine* 163 (2023): 107235.

60. J. Gui, T. Chen, J. Zhang, et al., "A Survey on Self-Supervised Learning: Algorithms, Applications, and Future Trends," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 46 (2024): 9052–9071.

61. R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-Cam: Visual Explanations From Deep Networks via Gradient-Based Localization," in *Proceedings of the IEEE International Conference on Computer Vision* (IEEE, 2017).

Supporting Information

Additional supporting information can be found online in the Supporting Information section.