Received: 15 April 2025 | Revised: 19 September 2025 | Accepted: 1 April 2026 | Published online: 19 May 2026

RESEARCH ARTICLE

Artificial Intelligence and Applications
2026, Vol. 00(00) 1–17
DOI: 10.47852/bonviewAIA52025938

BON VIEW PUBLISHING

Step-by-Step Approach to Design Image Classifiers in AI: An Exemplary Application of the CNN Architecture for Breast Cancer Diagnosis

Ahamadullah Lohani[1], Bhupesh Kumar Mishra[1,*], Kenneth Y. Wertheim[1] and Temitayo Matthew Fagbola[1]

[1] *Centre of Excellence for Data Science, Artificial Intelligence, and Modelling (DAIM), University of Hull, UK*

Abstract: Convolutional neural networks (CNNs) are commonly applied for image classification, but there is no standard protocol to facilitate comparison and synergy. This paper presents the first attempt at a step-by-step protocol for these purposes, exemplified by the problem of classifying ultrasound images for breast cancer diagnosis. Following this protocol, three datasets—Breast Ultrasound Image Dataset (BUSI), Breast Ultrasound Image (BUI), and Ultrasound Breast Images for Breast Cancer (UBIBC)—were used to build custom CNNs and fine-tune pre-trained CNNs by transfer learning. Then, they were optimized by data augmentation techniques, including random cropping, flipping, shifting, shearing, rotation, and zooming. Hyperparameters (batch size, learning rate, dropout rates, optimizer, and more) were tuned in a grid search in combination with learning rate scheduling and early stopping. Following these, ensemble modeling is also applied as a part of protocol and hence fusion-data. Cross-dataset evaluations were further conducted, where BUSI was used for training/validation and UBIBC for independent testing, and vice versa, to assess robustness and generalization. The obtained results indicate that the custom CNN and VGG19 (Visual Geometry Group 19-layer CNN) are most suitable for this problem. The custom sequential model achieved the highest performance level with an accuracy of 92%, precision of 93%, recall of 92%, F1-score of 92%, and area under the receiver operating characteristic curve of 99%. Employing the step-by-step approach not only results in a higher accuracy performing CNN-based classifier but also results in justifiable and resilient conclusions regarding image classification tasks to enhance the robustness and generalization capabilities of CNN-based classifiers. Beyond medical image classification tasks, the step-by-step approach offers a structured methodology that can enhance reproducibility, comparability, and clinical applicability classification tasks. By following this approach, researchers participating in different projects can produce comparable results, thus facilitating knowledge transfer and model reuse.

Keywords: convolutional neural network (CNN), deep learning (DL), image classification, breast cancer, ultrasound images, transfer learning (TL), data augmentation (DA)

1. Introduction

Artificial intelligence (AI) is rapidly evolving and integrates computational modeling, high-performance computing, and machine learning with big data records. The advent of convolutional neural networks (CNNs) has significantly enhanced these capabilities, leading to a surge in interest from different research communities, including medical diagnosis [1]. Furthermore, transfer learning (TL) has extended CNNs' applicability to scenarios with limited data, leveraging pre-trained models such as Residual Network (ResNet), U-Network (UNet), and VGGNet [2]. Although the basic CNN architecture is well-established, there are many variants of it with the publicly available pre-trained models that enable TL, as has been observed in a systematic review of 425 studies [3]. In addition to the existence of diverse pre-trained CNNs, many other refinement techniques are used to enhance the model performance, including the number of convolutional layers and their dimensions, fine-tuning layers and hyperparameters [4, 5], right optimizer from the many available options (like Adam, RMSprop, Nadam, and AdaGrad), regularization and model generalization, and overfitting prevention [6]. Refining a backbone architecture (like adding convolutional layers) and tuning its hyperparameters have also been used for optimizing a CNN model. Likewise, different types of data augmentation (DA) techniques [7, 8] have also been employed to improve model robustness. Nanni et al. [7] performed a comparative study of 11 augmentation techniques on four small imaging datasets to reduce overfitting. In their model, CNN were trained

***Corresponding author:** Bhupesh Kumar Mishra, Centre of Excellence for Data Science, Artificial Intelligence, and Modelling (DAIM), University of Hull, UK. Email: Bhupesh.Mishra@hull.ac.uk

were trained after DA and were merged into ensemble models to enhance accuracy.

Over the years, different alternatives have been presented for medical image classification as an exemplifying application of the CNN architecture [9]. In the pool of miscellaneous alternatives, the adjustment in CNN models to improve the image classification task is confounding several approaches on CNNs for breast cancer detection, such as Pacal et al. [10], who utilized CNNs to classify ultrasound breast images, highlighting the advantages of simple DA and TL, with the vision transformer model emerging as the top performer. Reenadevi et al. [11] employed a deep ResNet-152 model on histopathology images, achieving high accuracy. Similarly, Salama et al. [12] proposed a CNN-based framework for breast cancer segmentation and classification in mammograms, using the Mammographic Imaging Analysis Society (MIAS), Digital Database for Screening Mammography (DDSM), and Curated Breast Imaging Subset of DDSM (CBIS-DDSM) datasets. Al-Dhabyani et al. [13] demonstrated the effectiveness of Generative Adversarial Network (GAN)-based DA in classifying breast cancer from ultrasound images, with Neural Architecture Search Network (NASNet). Saber et al. [14] used TL of pre-trained CNNs for breast cancer detection and classification in mammograms, with VGG16 outperforming existing models after preprocessing and DA. Kumar et al. [15] proposed an approach for ultrasound image classification, achieving higher accuracy with a combination of DA, TL with BreastNet18, and a Cubic Support Vector Machine (SVM). Castro-Tapia et al. [16] explored multi-class breast cancer classification in mammograms using GoogLeNet, achieving superior performance across various metrics. Wakili et al. [17] applied TL with DenseNet for histopathology image classification, showcasing the effectiveness of CNNs and TL. Arooj et al. [4] customized AlexNet using TL and DA for ultrasound and histopathology images, achieving high accuracy. Similarly, Ayana et al. [18] examined TL approaches for ultrasound image diagnosis, emphasizing superior performance over non-transfer methods with various preprocessing techniques.

While applying a CNN-based model for image classification, one may overlook some of the many model enhancement techniques or their combinations in an unorganized way, making the effort inefficient. A common framework is a justified need to find the best approach for image classification using CNN. A common framework can reduce the risk and realize the opportunity by facilitating comparison between methods and results, contributing to quality control, and upscaling. Sijie et al. [19] and Cui et al. [20] have argued for establishing and using a common approach in AI-assisted image classification that can arrange refinement techniques in a step-by-step manner to get the most out of each CNN architecture. Similarly, a common problem encountered by AI learners and researchers is that they stop experimenting with alternative architectures, refinement techniques, and DA techniques after achieving decent results. The existence of diverse evaluation metrics and visualizations often confuses AI researchers, too. Hence, it appears that a common mistake for AI researchers is to rely on a narrow range of or maybe even just one metric or visualization for evaluation. Unlike traditional disciplines like physics and economics, AI does not have a clear and standardized framework for image-based classification using CNN. Considering these, this study aims to propose a comprehensive approach as a standardized framework for optimizing the use of CNNs in image classification.

The approach proposed in this paper encourages image classification tasks to move in this direction by highlighting how a standardized framework aids in selecting the most suitable CNN architecture and refinement techniques for specific tasks. This study emphasizes that an approach can enhance image classification tasks by providing a clear framework with the exemplary application of breast cancer detection. In other words, the study contributes to the domain of image classification using CNN in several key ways: (i) It highlights the need for a standardized approach to reduce redundancy and foster synergy in CNN-based image classification. (ii) It offers a detailed framework for selecting and optimizing CNN architectures, including refinement techniques and DA methods. (iii) It also provides insights into how such a standardized approach can enhance AI education by clarifying objectives and improving the learning experience for students and educators. This approach could also serve as a roadmap for future advancements in image classification and the fast-growing field of image-based computational applications in healthcare. Furthermore, it may be adapted for other classification tasks, ensuring similar levels of accuracy and methodological rigor. The experimental findings emphasize the importance of selecting appropriate tuning strategies in image classification tasks to improve the robustness and generalization of CNN-based classifiers.

The rest of the paper is organized as follows: the Materials and Methods used in this study are presented in Section 2, the results of the adopted models are discussed in Section 3, the discussion on the models' performance is provided in Section 4, and the conclusions of the study are summarized in Section 5.

2. Materials and Methods

This study introduces a step-by-step guide for building a CNN-based image classification system using exemplary breast cancer detection. The study highlights the potential of deep learning (DL) and TL in improving the accuracy and efficiency of image classification for cancer detection, proving their vital role in early diagnosis and treatment. The adopted computational experimentation includes building, refining, and experimenting with custom and pre-trained CNN models, incorporating TL, DA, hyperparametric tuning, and ensemble modeling. These models are tested on datasets with varying sizes and levels of clarity and purity in nuanced ways and hence exposed to seen or unseen (seen data refers to the dataset(s) used during training, validation, and testing phases, while unseen data refers to dataset(s) not used in training), which are employed to evaluate model performance and generalization on new, previously unencountered images, ensuring robust evaluation metrics and visualizations. Testing unseen data enables assessment of the model's generalization capabilities and robustness. This approach ensures that evaluation metrics and visualizations reflect the model's performance on both familiar and novel inputs.

In this study, the three datasets—Breast Ultrasound Image Dataset (BUSI), Breast Ultrasound Image (BUI), and Ultrasound Breast Images for Breast Cancer (UBIBC)—are used, where UBIBC is a large dataset (with more than 9000 images) with high-quality images in PNG (Portable Network Graphics) format, BUSI is a medium-sized dataset with 780 images in PNG format, and BUI is a smaller dataset with 250 slightly blurred images in Bitmap (BMP) format. A standardizing criterion was applied such that all images were converted to a uniform resolution of 224 × 224 pixels, standardized to grayscale, and saved in PNG format to ensure consistency. Class imbalances were addressed using the Synthetic Minority Oversampling Technique (SMOTE), which generates synthetic minority class samples by interpolating existing examples, thereby balancing the dataset. Pixel values were normalized using min–max rescaling, where intensity values were scaled to a [0, 1] range to ensure uniformity across images

and improve model convergence. Furthermore, DA methods were also employed to enhance data diversity and model robustness. While multiple imaging approaches, such as Magnetic Resonance Imaging (MRI) and X-rays, have been used for breast cancer detection, this study specifically focuses on ultrasound imaging, as ultrasound images are particularly useful for distinguishing between cysts and solid masses in dense breast tissue, making it a preferred choice in specific scenarios. In addition, several standardized evaluation metrics were also used for model performance, including accuracy, precision, recall, F1-score, and AUC (area under the receiver operating characteristic curve). Accuracy measures the overall correctness of classifications, while precision reflects the proportion of true positives (TP) among predicted positives. Recall focuses on detecting TP, and the F1-score balances precision and recall, especially in imbalanced datasets. Similarly, AUC evaluates the model's ability to differentiate between classes. Visualizations like confusion matrices and Receiver Operating Characteristic (ROC) curves complement these metrics, ensuring clear and standardized interpretation for both experts and non-experts.

Furthermore, a series of breast cancer image classifiers were built systematically with custom and pre-trained CNN architectures, with and without TL. To sum up, the study implemented a thorough journey of exploring different convolution blocks, frozen layers, normalization techniques, data balancing techniques, DA methods, hyperparameter tuning (activation functions, optimizers, batch sizes, and learning rates), and regularization methods (dropout and early stopping mechanisms), as well as different dataset sizes. Figure 1 illustrates the steps in the approach to building exemplary image classifiers. The process begins by developing base models from scratch without utilizing TL. These models were first trained with the BUSI dataset without TL, and then TL was employed by importing pre-trained models and fine-tuning them. DA techniques were subsequently applied to diversify the training dataset. After this, the step was followed by hyperparameter tuning and layer tuning to optimize the models. Ensemble models were implemented using a stacked approach. In the first attempt at ensembling, features extracted from VGG19 were combined with a custom sequential CNN, but performance was limited. In the next step of ensembling, a more complex ensemble was then created by concatenating the feature outputs of VGG19 and InceptionResNetV2, which were fed into fully connected layers for final classification.

In addition, a combination of BUSI and BUI datasets and BUSI and UBIBC datasets was also used in this work, where models were first tested using the same dataset by splitting into training, validation, and testing data. This allowed performance evaluation of the model from the same dataset. Following this, the models were further tested using unseen data and evaluated for how well the models were generalized to new, previously unencountered data. In the next step, models were tested on a larger training dataset and hence the evaluation of their performance. With these steps, the presented work is structured as a framework that provides a systematic step-by-step approach to CNN-based image classifier development and evaluation, ensuring a range of alternative techniques with the available datasets in an image classification task to achieve optimal performance in a task-specific and context-specific manner.

2.1. Data preparation

Different datasets with different numbers of image data are used in this study, as shown in Table 1, which are also illustrated in Figures 2, 3, and 4 with random sample images from different datasets that have been used in this study. To build the CNN models, each dataset was systematically partitioned into training, validation, and testing subsets to prevent data overlapping across these stages. Additionally, in some experiments, distinct datasets were used for training, validation, and testing to assess the robustness of model performance based on unseen data (data belonging to different datasets), mitigating overfitting and enhancing overall performance.

Figure 1
Conceptual workflow of the framework

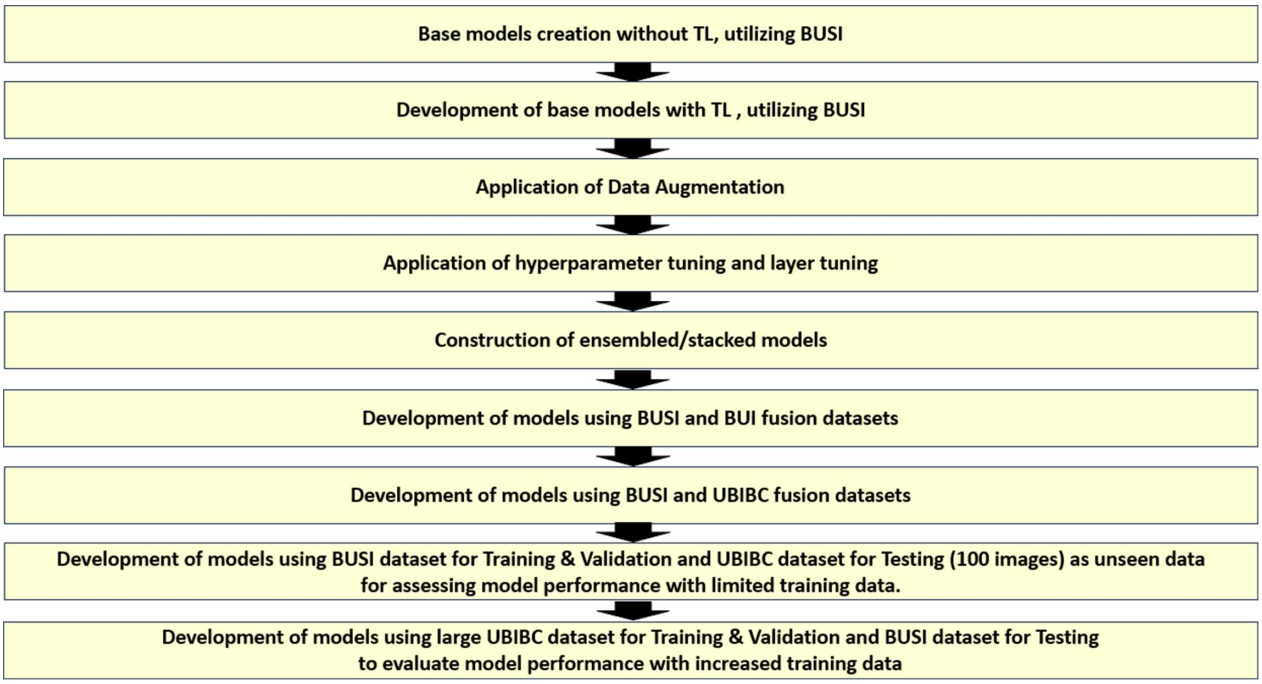


Table 1
Different datasets used in the research study

Dataset	Images		Class		
	Total	Format	Benign	Malignant	Normal
BUSI -Breast UltraSound Images	780	PNG	437	210	133
BUI - Breast Ultrasound Images	250	BMP	100	150	–
UBIBC - Ultrasound Breast Images for Breast Cancer	9016	PNG	4574	4442	–

Figure 2
Sample images of BUSI—Breast UltraSound Images

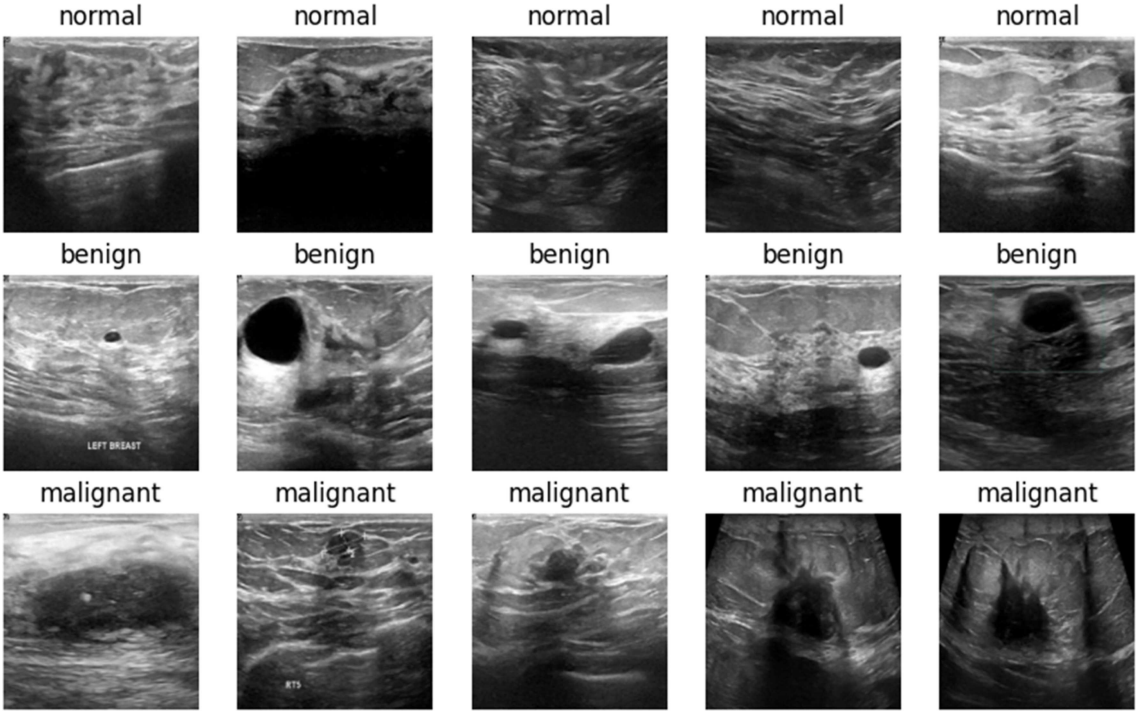


Figure 3
Sample images of BUI—Breast Ultrasound Images

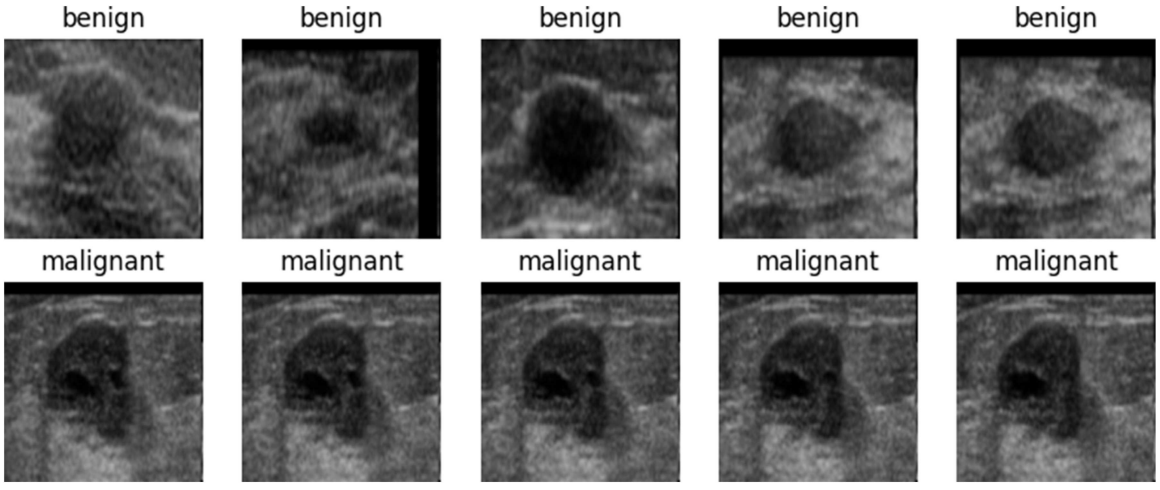
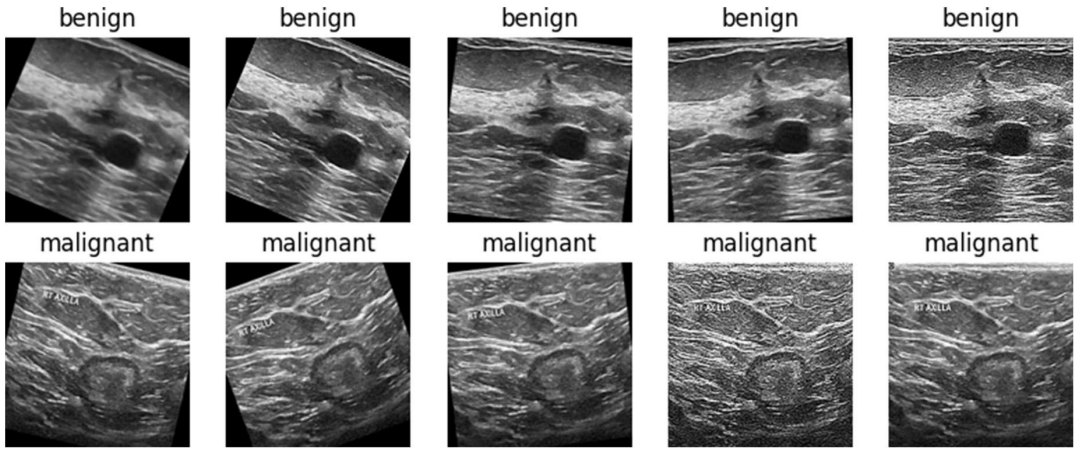


Figure 4
Sample images of UBIBC—Ultrasound Breast Images for Breast Cancer



2.2. Data preprocessing

Data preprocessing was carried out systematically to prepare the datasets for model development. Image formats were standardized by converting all images to grayscale, resizing them to 224 × 224 pixels, and saving them in PNG format to ensure consistency across datasets. To address class imbalance, the SMOTE was applied. Pixel values were normalized to the range [0, 1] using min–max scaling (dividing raw pixel values by 255) to facilitate faster model convergence. Finally, DA techniques, specifically random cropping, horizontal/vertical flipping, horizontal/Vertical shifting, shearing, rotation, and zooming, were applied during training to expand the dataset and introduce diversity to the training data, thereby improving model robustness.

2.3. Model development

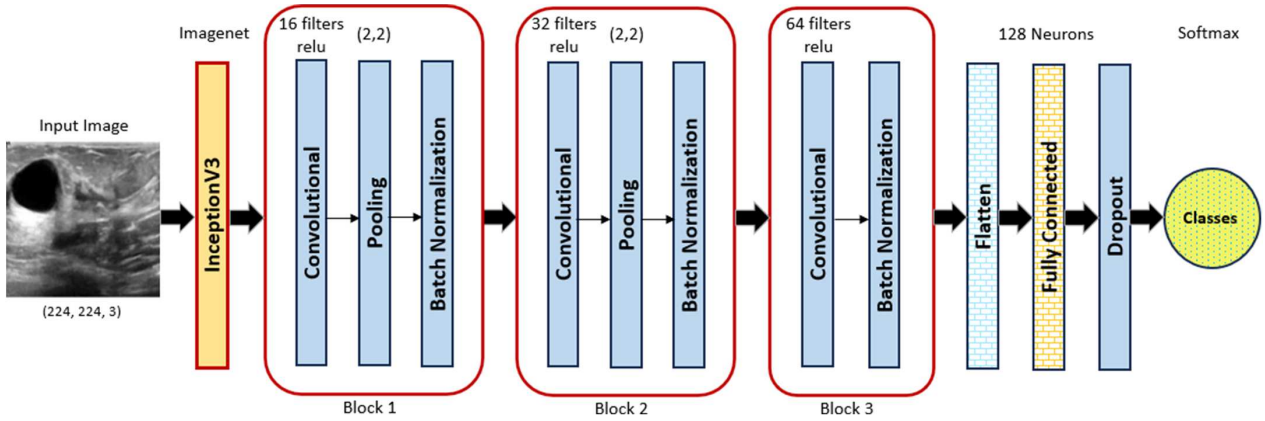
Out of the many CNN architectures available in the public domain, a shortlist was devised for breast cancer classification based on the recommendations in the scholarly articles [3, 10, 11, 19, 21]. Custom and pre-trained CNN architectures with and without TL were used to build a series of models, which were enhanced methodically with architectural variations (as shown in Figures 5 and 6), normalization techniques, and hyperparameter tuning techniques. Pixel values were normalized using min–max scaling (dividing each raw pixel value by 255) to map values into the [0, 1] range. Hyperparameter tuning was carried out through a combination of optimization and manual adjustment of parameters including optimizers (Stochastic Gradient Descent (SGD), Nadam, AdaGrad), activation functions, dropout rate (0.5, 0.6), batch size (16, 32, 128), L2 regularization (0.01), and dynamic learning rate scheduling (reducing the learning rate by a factor of 0.5 if validation loss did not improve for 5 consecutive epochs). The selected pre-trained CNN models were initialized with ImageNet pre-trained weights, and their dense custom layers were initialized using GlorotUniform (Xavier uniform) to optimize training efficiency and mitigate gradient-related issues. Optimizers were used with the categorical cross-entropy loss as the objective function. Class weights were incorporated using the "balanced" option to address imbalanced datasets. The resulting weights were stored in a dictionary in a class-dependent manner, ensuring that minority classes would be oversampled during model training.

Classification models were trained using alternative strategies involving distinct datasets, including hybrid datasets created by fusion. At first, the BUSI and BUI datasets were fused before being meticulously split into training (70%), validation (15%), and test (15%) subsets [10]. A similar approach was applied to the fusion of the BUSI and UBIBC datasets. In one qualitatively different case, not involving hybrid datasets, the BUSI dataset was split into two subsets for training and validation, while selected UBIBC images were used for testing. Similarly, models were trained and validated on a large dataset (UBIBC) and evaluated against smaller datasets (BUSI). To prevent overfitting, early stopping was used during model training by monitoring each model's validation loss and accuracy to gain insights into its learning dynamics and terminating its training when it was observed to diverge from its training loss. A learning rate scheduler was utilized to adjust (decrease) the learning rate by a factor of 0.5 if the validation loss stagnated for 5 consecutive epochs. Each trained model was tested on a test dataset and in terms of a spectrum of key metrics, including testing accuracy, precision, recall, F1-score, and AUC-ROC curve [6]. Confusion matrices (a tabular dataset that presents the performance of a classification model by comparing predicted values against actual values) were also built to supplement these individual metrics by providing a holistic assessment of the model's performance. Figure 5 shows the sequential model architecture comprising a pre-trained InceptionV3 base model followed by additional layers for feature extraction and classification. The additional layers begin with two convolutional layers with 16 and 32 filters, respectively. Their outputs pass through Rectified Linear Unit (ReLU) activation functions (a nonlinear function whose outputs are the input directly if it's positive and outputs zero for any negative input, mathematically defined as equation 1, average pooling layers, and batch normalization function. Batch normalization makes sure that the outputs of each layer stay steady as the model learns by subtracting the batch mean and dividing by the batch standard deviation, followed by scaling and shifting using learnable parameters. For the batch normalization, the mean and standard deviation are calculated using equations 2 and 3, respectively, and then normalized using equation 4.

$$\text{ReLU: } f(x) = \max(0,\ x) \tag{1}$$

$$Mean\ (\mu) = \frac{1}{m}\sum_{i=1}^{m} x_i \tag{2}$$

Figure 5
Architecture of Keras sequential model with transfer learning



$$\textit{Standard Deviation } (\sigma) = \sqrt{\left(\frac{\sum_{i=1}^{m}(x_i - \mu)^2}{N}\right) + \epsilon} \quad (3)$$

where m is the number of examples in the mini batch, x_i is a single example in the batch, N is the number of data points, and $\in$ is a small constant.

$$\text{Normalize: } \hat{x} = \left(\frac{x_i - \mu}{\sqrt{\sigma^2}}\right) \quad (4)$$

Following this, scaling and shifting are applied with a learnable parameter, gamma (γ), and shifted by another learnable parameter, beta (β), using equation 4

$$y_i = \gamma\hat{x} + \beta \quad (5)$$

After this, an additional convolutional layer with 64 filters, followed by a flattening layer and a dense layer comprising 128 units with ReLU activation and dropout regularization, which randomly deactivates neurons during training to prevent overfitting by reducing co-adaptation among neurons and mathematically represented as equation 6, was used. The final layer simply comprises two units with a softmax activation, which converts the outputs into class probabilities for multi-class classification problems and is mathematically represented as equation 7. In addition to dropout regularization, some of the dense layers are regularized through L2 regularization with a coefficient of 0.01 to prevent overfitting by adding a penalty to the model's cost function, mathematically represented as equation 8. Figure 6 shows the architecture of a general predefined CNN. In each specific implementation of this general architecture, the model's pre-trained weights were derived from the ImageNet dataset and fine-tuned by freezing different layers, for example, freezing the 10 layers before the last layer.

Dropout regularization:

$$m_i \sim \text{Bernoulli}(p), \quad \tilde{h}_i = (h_i \times m_i)/p, \quad E[\tilde{h}_i] = h_i \quad (6)$$

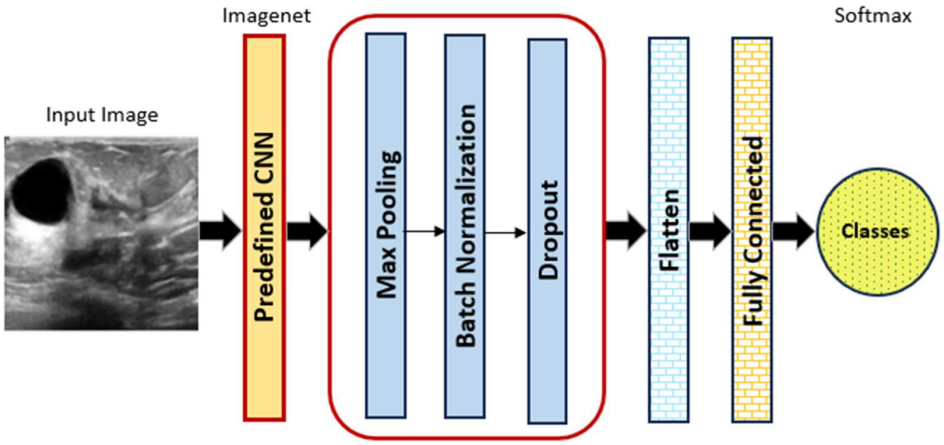
where m_i is the dropout mask sampled from a Bernoulli distribution with probability p and h_i represents the neuron activation.

Softmax function:

$$\sigma(z)_i = ez_i/\Sigma_j\, ez_j,\ i = 1, \ldots, K \quad (7)$$

where e is the base of the natural logarithm (Euler's number), z_i is the i^{th} element of the input vector, and K is the number of elements in the input vector.

Figure 6
Architecture of predefined CNN models with transfer learning



L2 regularization (weight decay):

$$\Omega_{12}(W) = (\lambda/2) \times ||W||^{2}{}_{2} = (\lambda/2) \times \sum_{i} w_{i}^{2}$$
$$L = L_{0} + (\lambda/2) \times ||W||^{2}{}_{2} \quad (8)$$

where λ is the regularization coefficient, L_0 is the original loss function, and W is the model's weights. These weights are the coefficients that determine the relationship between the input features and the model's output.

3. Experimental Analysis

Following the conceptual framework described in Section 2, a series of machine learning experiments was conducted in alignment with the presented method to build, optimize, and evaluate the exemplary application of breast cancer image classifiers. These experiments, with different alternatives, were conducted for image classification tasks, moving in this direction by the importance of how a standardized step-by-step approach helps in selecting the most suitable CNN architecture and enhancement techniques for domain-specific tasks. For the experimental analysis, three datasets, along with the combination DA, were used for model training. For the evaluation of the model, performance measures—accuracy (measures the proportion of correct predictions made by a model out of all the predictions it has made), precision (measures the accuracy of a model's positive predictions), recall (measures how many of the actual positive cases a model correctly identifies), F1-score (measures the performance of a classification model by calculating the harmonic mean of its precision and recall), and AUC (probability that a model will rank a randomly chosen positive example higher than a randomly chosen negative example)—are calculated for each model and hence analyzed.

3.1. Experiment 1: Base models with and without transfer learning

At first, 10 base models were created without TL, encompassing architectures such as Sequential in Keras (custom) and predefined models (shortlisted based on recommendations in the articles) [3, 10, 11, 19, 21]. Simultaneously, another 10 base models were developed based on the same architectures with TL, with the weights being initialized with the "imagenet" setting for improved feature extraction, as listed in Table 2. For the analysis, the first 10 models without TL performed suboptimally compared to their counterparts with TL. The base models, without TL, were not learning effectively as the loss and accuracy values did not improve over epochs. On the other hand, the models with TL improved consistently in terms of training loss, indicative of effective learning and generalization to unseen data. This shows that leveraging pre-trained weights helped the second set of 10 models learn from the training dataset effectively, aligning with expectations. Sequential, EfficientNetB3, VGG19, ResNet152, InceptionV3, and InceptionResNet2 models with TL were notably performant in this experiment.

The evaluation metrics—precision, recall, F1-score, and accuracy—were analyzed for each model for each class (0: "normal," 1: "benign," 2: "malignant") as listed in Table 2. The table shows that TL has built efficient classifiers, particularly evident in precision, recall, and F1-score metrics. For instance, the VGG19, InceptionV3, InceptionResNetV2, and Sequential in Keras models with TL achieved levels of accuracy of 88%, 85%, 85%, and 81%, respectively.

3.2. Experiment 2: Application of data augmentation techniques

In the next stage, various DA techniques were utilized as compatibility listed in Table 3, where DA combinations were shortlisted after reviewing different articles [2, 4, 5, 8, 10–12, 16–18, 22–24], as summarized in Table 3. Horizontal and vertical flips, rotation, shear, zoom, width and height shift, and cropping were shortlisted due to their efficacy in the reviewed articles. Four combinations of them were applied to Sequential in Keras, EfficientNetB3, VGG19, ResNet152, InceptionV3, and Inception-ResNet2, which were the top performers in the first experiment. Among these combinations, horizontal and vertical flip, horizontal and vertical shift, shear, rotation, zoom, and cropping achieved high accuracy, as summarized in Table 4.

3.3. Experiment 3: Bayesian hyperparameter tuning

Bayesian optimization enabled hyperparameter tuning affecting the models systematically, tailored by the findings of Balaha et al. [21]. The experiment identified contributing factors to model robustness are two activation functions (ReLU and softmax), L2 regularization with a coefficient of 0.01, batch normalization (16, 32), early stopping, dropout (0.5), convolution blocks (16, 32, 64 filters), and use of dynamic learning rate scheduler to reduce the default learning rate (0.001) by a factor of 0.5. Notably, different TL-based models worked well with different optimizers (AdaGrad, SGD, Nadam), emphasizing the nuanced impact of hyperparameter tuning on different architectures. In addition, different patterns of frozen layers with selective freezing, such as freezing the last 10 layers and freezing the penultimate layer only, were also experimented with [3, 5]. However, this did not improve the results; in some cases, it had a counterproductive effect. The analysis, presented in Table 5, supplemented by Figure 7, shows that Nadam is an effective optimizer for Sequential in Keras, InceptionV3, and InceptionRes-NetV2 models, while AdaGrad performed well for VGG19, EfficientNetB3, and ResNet152.

3.4. Experiment 4: Stacked (ensemble) models

In the next step, the effectiveness of stacked models for image classification was analyzed. Initial attempts involved stacking VGG19, used for feature extraction, on Sequential in Keras, but suboptimal results were observed. Subsequently, a more complex model was devised by stacking VGG19 and InceptionResNetV2, which concatenated their extracted features before feeding them to a final series of fully connected layers for classification. However, this sophisticated approach also did not achieve significantly better results. The steps highlighted that there have been potential difficulties in exploiting an ensemble approach with a relatively small dataset.

3.5. Experiment 5: Splitting BUSI/BUI fusion dataset into training, validation, and testing subsets

Furthermore, in the next step, a comprehensive approach was implemented to enhance the performance of DL models by combining BUSI and BUI datasets and splitting the resulting BUSI/BUI fusion with diverse datasets into training (70%), validation (15%), and test (15%) subsets. The analysis has shown that models based on the VGG19 and InceptionResNetV2 architectures were fine-tuned by TL and empowered by DA, hyperparameter tuning, and class balancing with class weights,

Table 2
Base models with and without transfer learning

Image size: 224 x 224			Data Augmentation: None					Balancing: SMOTE					Epochs: 30					
Dataset split: 70%-15%-15%			Normalise: Yes					Optimiser: SGD										
			Precision					Recall					F1-Score					
	Model	Overfitting / Underfitting?	0	1	2	MA	WA	0	1	2	MA	WA	0	1	2	MA	WA	Accuracy
Without Transfer Learning	Sequential in Keras	OF	0.25	0.67	0.9	0.61	0.66	0.05	0.94	0.58	0.52	0.69	0.08	0.78	0.71	0.52	0.64	69%
	EfficientNetB0	UF	0.17	0.0	0.0	0.06	0.03	1.0	0.0	0.0	0.33	0.17	0.29	0.0	0.0	0.1	0.05	17%
	EfficientNetB3	UF	0.0	0.0	0.26	0.09	0.07	0.0	0.0	1.0	0.33	0.26	0.0	0.0	0.42	0.14	0.11	26%
	VGG16	UF	0.0	0.56	0.0	0.19	0.32	0.0	1.0	0.0	0.33	0.56	0.0	0.72	0.0	0.24	0.41	56%
	VGG19	UF	0.06	0.0	0.2	0.09	0.06	0.05	0.0	0.65	0.23	0.18	0.06	0.0	0.3	0.12	0.09	18%
	ResNet50	UF	0.0	0.58	0.67	0.42	0.51	0.0	0.95	0.19	0.38	0.59	0.0	0.72	0.3	0.34	0.49	59%
	ResNet101	UF	0.0	0.58	0.75	0.44	0.53	0.0	0.95	0.19	0.38	0.59	0.0	0.72	0.31	0.34	0.49	59%
	ResNet152	OF	0.0	0.68	0.4	0.36	0.49	0.0	0.62	0.74	0.45	0.55	0.0	0.65	0.52	0.39	0.51	55%
	InceptionV3	UF	0.0	0.56	0.0	0.19	0.32	0.0	1.0	0.0	0.33	0.56	0.0	0.72	0.0	0.24	0.41	56%
	InceptionResNetV2	UF	0.0	0.6	0.62	0.41	0.51	0.0	0.97	0.16	0.38	0.59	0.0	0.74	0.26	0.33	0.49	59%
With Transfer Learning	Sequential in Keras	No	0.75	0.8	0.88	0.81	0.81	0.6	0.92	0.71	0.74	0.81	0.67	0.86	0.79	0.77	0.81	81%
	EfficientNetB0	UF	0.14	0.67	0.38	0.39	0.5	0.35	0.27	0.48	0.37	0.34	0.2	0.39	0.42	0.34	0.36	34%
	EfficientNetB3	OF	0.47	0.92	0.79	0.73	0.81	0.9	0.7	0.74	0.78	0.74	0.62	0.79	0.77	0.73	0.76	74%
	VGG16	No	1.0	0.93	0.68	0.87	0.88	0.75	0.86	0.9	0.84	0.85	0.86	0.9	0.78	0.84	0.86	85%
	VGG19	No	0.81	0.94	0.82	0.86	0.88	0.85	0.88	0.9	0.88	0.88	0.83	0.91	0.86	0.87	0.88	88%
	ResNet50	OF	0.36	0.41	0.22	0.33	0.35	0.25	0.18	0.52	0.32	0.28	0.29	0.25	0.3	0.28	0.27	28%
	ResNet101	OF	1.0	0.55	0.0	0.52	0.48	0.1	0.89	0.0	0.33	0.52	0.18	0.68	0.0	0.29	0.42	52%
	ResNet152	OF	0.5	0.56	0.25	0.44	0.47	0.05	0.94	0.03	0.34	0.55	0.09	0.7	0.06	0.28	0.43	55%
	InceptionV3	No	0.75	0.87	0.86	0.83	0.85	0.75	0.91	0.77	0.81	0.85	0.75	0.89	0.81	0.82	0.85	85%
	InceptionResNetV2	No	0.9	0.86	0.79	0.85	0.84	0.9	0.89	0.71	0.83	0.85	0.9	0.87	0.75	0.84	0.84	85%

Normalise: Normalising training, validation, and test data by dividing by 255, resulting in values in the range between 0 and 1
Dataset Split %: 70% Training data; 15% Validation data; 15% Test data
SMOTE: Synthetic Minority Over-sampling Technique
OF: Overfilling; UF: Underfitting
MA - Macro Average; WA - Weighted Average
SGD: Stochastic Gradient Descent
Class: 0 - Normal; 1- Benign; 2 - Malignant

Table 3
Data augmentation techniques

Horizontal flip	Vertical flip	Horizontal shift	Vertical shift	Rotate	Shear	Zoom	Crop	Resize	CLAHE	Mixing	Erasing	Contrast	Color change	Noice injection	GAN	References	Images type
Y	Y			Y												Pacal [10]	BUSI—Ultrasound images
Y				Y			Y									Reenadevi et al. [11]	BreakHis—Histopathology images
				Y												Salama et al. [12]	MIAS, DDSM, CBIS-DDSM—Mammogram images
															Y	Al-Dhabyani et al. [13]	Ultrasound images
Y	Y							Y								Saber et al. [14]	MIAS—Mammogram images
				Y					Y							Castro-Tapia et al. [16]	MIAS, INbreast—Mammogram images
		Y	Y	Y		Y										Arooj et al. [4]	Ultrasound and histopathology images
Y	Y			Y			Y									Abdou [23]	X-ray, MRI, CT, ultrasound, and photography images
Y	Y			Y	Y	Y										Kathamuthu et al. [22]	CT scan images
Y	Y				Y		Y			Y	Y				Y	Yang et al. [24]	Non-medical images: CIFAR-10, CIFAR-100, SVHN, COCO
Y	Y			Y			Y						Y	Y		Khosla et al. [8]	Non-medical images: CIFAR-10, CIFAR-100, MNIST, SVHN
Y	Y			Y	Y							Y				Korzhebin et al. [5]	Non-medical images (total 40)

Table 4
Models' performance measures with transfer learning and data augmentation with the BUSI dataset

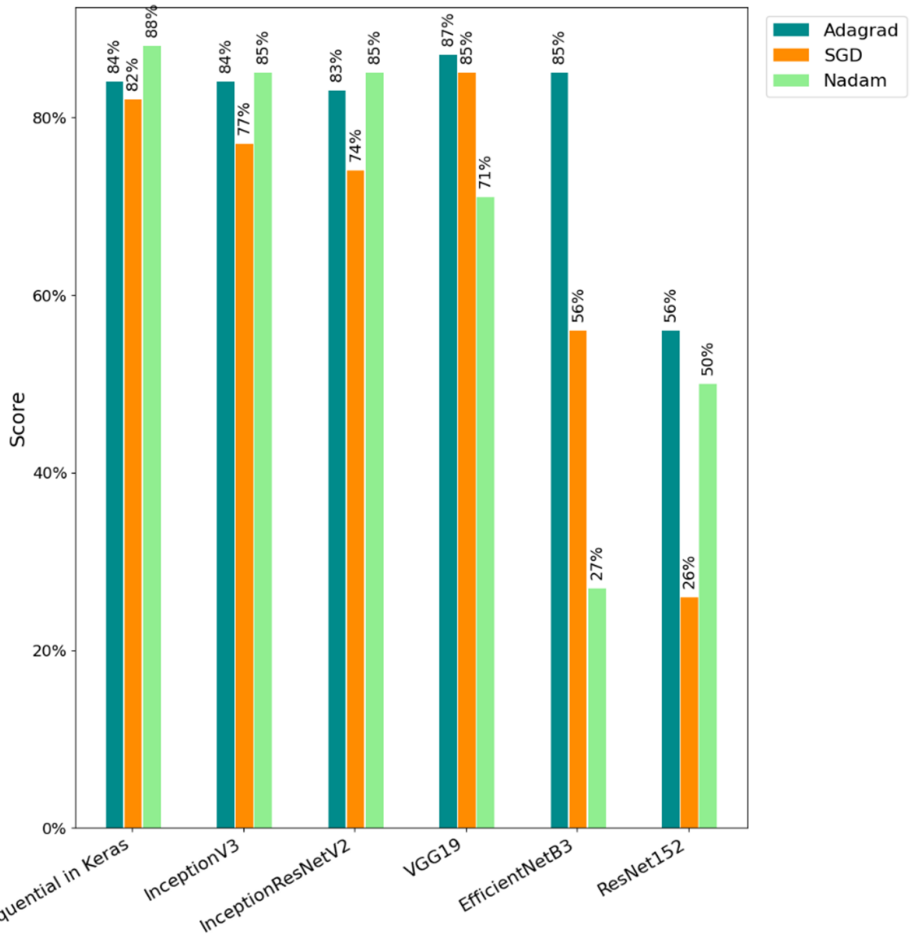
Normalise: Yes			Balancing: SMOTE					Optimiser: SGD					Epochs: 30					
			Precision					Recall					F1-Score					
	Model	Data Augmentation	0	1	2	MA	WA	0	1	2	MA	WA	0	1	2	MA	WA	Accuracy
With Transfer Learning & Data Augmentation	Sequential in Keras	Hf, R, C	0.82	0.94	0.82	0.86	0.88	0.90	0.88	0.87	0.88	0.88	0.86	0.91	0.84	0.87	0.88	88%
		H&Vf , S, C	0.78	0.87	0.77	0.81	0.83	0.90	0.83	0.77	0.84	0.83	0.84	0.85	0.77	0.82	0.83	83%
		H&Vf , R, S, Z	0.69	0.88	0.88	0.82	0.85	0.90	0.86	0.74	0.84	0.84	0.78	0.87	0.81	0.82	0.84	84%
		H&Vf , H&Vs, S, R, Z, C	0.76	0.91	0.92	0.86	0.89	0.90	0.91	0.77	0.88	0.88	0.84	0.91	0.84	0.87	0.88	88%
	EfficientNetB3	Hf, R, C	0.00	0.57	0.00	0.19	0.32	0.00	1.00	0.00	0.33	0.56	0.00	0.73	0.00	0.24	0.41	56%
		H&Vf , S, C	0.00	0.64	0.42	0.35	0.47	0.00	0.70	0.61	0.44	0.56	0.00	0.67	0.50	0.39	0.51	56%
		H&Vf , R, S, Z	0.37	0.72	0.57	0.55	0.62	0.65	0.67	0.39	0.57	0.59	0.47	0.69	0.46	0.54	0.59	59%
		H&Vf , H&Vs, S, R, Z, C	0.43	0.74	0.61	0.59	0.65	0.50	0.50	0.35	0.57	0.66	0.47	0.79	0.45	0.57	0.64	66%
	VGG19	Hf, R, C	0.64	0.78	0.86	0.76	0.78	0.45	0.88	0.81	0.71	0.79	0.53	0.83	0.83	0.73	0.78	79%
		H&Vf , S, C	0.88	0.92	0.64	0.81	0.84	0.70	0.82	0.87	0.80	0.81	0.78	0.86	0.74	0.79	0.82	81%
		H&Vf , R, S, Z	0.12	0.62	0.00	0.25	0.37	0.20	0.80	0.00	0.33	0.49	0.15	0.70	0.00	0.29	0.42	49%
		H&Vf , H&Vs, S, R, Z, C	0.80	0.89	0.79	0.83	0.85	0.80	0.86	0.84	0.83	0.85	0.80	0.88	0.81	0.83	0.85	85%
	ResNet152	Hf, R, C	0.00	0.56	0.00	0.19	0.32	0.00	1.00	0.00	0.33	0.56	0.00	0.72	0.00	0.24	0.41	56%
		H&Vf , S, C	0.00	0.56	0.00	0.19	0.32	0.00	1.00	0.00	0.33	0.56	0.00	0.72	0.00	0.24	0.41	56%
		H&Vf , R, S, Z	0.00	0.56	0.00	0.19	0.32	0.00	1.00	0.00	0.33	0.56	0.00	0.72	0.00	0.24	0.41	56%
		H&Vf , H&Vs, S, R, Z, C	0.00	0.56	0.00	0.19	0.32	0.00	1.00	0.00	0.33	0.56	0.00	0.72	0.00	0.24	0.41	56%
	InceptionV3	Hf, R, C	0.76	0.86	0.73	0.78	0.81	0.65	0.82	0.87	0.78	0.80	0.70	0.84	0.79	0.78	0.80	80%
		H&Vf , S, C	0.88	0.86	0.84	0.86	0.86	0.70	0.91	0.84	0.82	0.85	0.78	0.88	0.84	0.83	0.85	85%
		H&Vf , R, S, Z	0.84	0.88	0.83	0.85	0.86	0.80	0.91	0.81	0.84	0.86	0.82	0.90	0.82	0.85	0.86	86%
		H&Vf , H&Vs, S, R, Z, C	0.88	0.85	0.89	0.87	0.87	0.70	0.94	0.81	0.82	0.86	0.78	0.89	0.85	0.84	0.86	86%
	InceptionResNetV2	Hf, R, C	0.69	0.91	0.78	0.79	0.84	1.00	0.77	0.81	0.86	0.82	0.82	0.84	0.79	0.82	0.82	82%
		H&Vf , S, C	0.78	0.90	0.76	0.82	0.84	0.90	0.82	0.84	0.85	0.84	0.84	0.86	0.80	0.83	0.84	84%
		H&Vf , R, S, Z	0.78	0.89	0.78	0.82	0.84	0.90	0.83	0.81	0.85	0.84	0.84	0.86	0.79	0.83	0.84	84%
		H&Vf , H&Vs, S, R, Z, C	0.82	0.89	0.78	0.83	0.85	0.90	0.85	0.81	0.85	0.85	0.86	0.87	0.79	0.84	0.85	85%

Hf: Horizontal flip; R: Rotate; C: Crop; S: Shear; Z: Zoom; H&Vf: Horizontal & Vertical flip; H&Vs: Horizontal &Vertical shift
Normalize: Normalizing training, validation, and test data by dividing by 255 resulting in values in the range between 0 and 1
Dataset Split %: 70% Training data; 15% Validation data; 15% Test data
Class: 0 -Normal; 1-Benign; 2 -Malignant
SMOTE: Synthetic Minority Over-sampling Technique
MA - Macro Average
SGD: Stochastic Gradient Descent
WA - Weighted Average

Table 5
Models' performance measures after tuning with the BUSI dataset

Optimizer	Sequential in Keras accuracy (%)	Inception V3 accuracy (%)	Inception-ResNetV2 accuracy (%)	VGG19 accuracy (%)	Efficient-NetB3 accuracy (%)	ResNet152 accuracy (%)
AdaGrad	84	84	83	87	85	56
SGD	82	77	74	85	56	26
Nadam	88	85	85	71	27	50

Figure 7
Optimizer selection based on the accuracy of models



which achieved a slightly higher level of accuracy at the cost of a diminished AUC score. Additionally, applying GlorotUniform (also known as Xavier uniform) weight initialization for the dense custom layers yielded a marginal accuracy boost in some models at the cost of a diminished AUC score.

3.6. Experiment 6: Fusion of BUSI and UBIBC datasets

3.6.1. Experiment 6(a): Splitting fusion dataset into training, validation, and testing subsets

A combined dataset created by the fusion of BUSI and UBIBC datasets was meticulously split into training (70%), validation (15%), and test (15%) subsets. Models constructed using Sequential in Keras, VGG19, InceptionV3, and InceptionResNetV2 exhibited better testing results in terms of precision, recall, F1-score, AUC, and accuracy. Notably, Sequential in Keras and VGG19 emerged as the top performers in this experiment. The results of this experiment are presented in Table 6, where the sequential model in Keras performed best.

3.6.2. Experiment 6(b): Using different datasets for training/validation and testing

In this setup, the BUSI dataset was used for training and validation, while UBIBC was only used for testing. The results, as listed in Table 7, demonstrated the models' robustness by generalizing to the unseen dataset (UBIBC) without difficulty. VGG19 achieved an accuracy of 82%, AUC scores of 93% for benign and malignant classes, and precision, recall, and F1-score of 84%, 82%, and 82%, respectively, as listed in Table 7.

Analyzing both the fusion, it has been observed that the hybrid dataset allowed the models to generalize and perform well

Table 6
Models' performance measures with transfer learning and data augmentation with BUSI and UBIBC datasets

Model	Optimizer	Precision		Recall		F1-Score		AUC		Accuracy(%)
		MA	WA	MA	WA	MA	WA	0	1	
Sequential in Keras	Nadam	0.95	0.96	0.96	0.96	0.95	0.96	0.99	0.99	96
VGG19	AdaGrad	0.93	0.94	0.94	0.94	0.93	0.94	0.99	0.99	94
InceptionV3	Nadam	0.89	0.91	0.91	0.90	0.89	0.90	0.98	0.98	90
Inception ResNetV2	Nadam	0.91	0.88	0.80	0.85	0.82	0.84	0.99	0.99	85
Combined BUSI (647 images) and UBIBC (446 original images) datasets and the total 1093 images split into training (70%), validation (15%), and test (15%) datasets.										

Table 7
Models' performance measures after tuning with BUSI and UBIBC datasets

Model	Optimizer	Precision		Recall		F1-Score		AUC		Accuracy(%)
		MA	WA	MA	WA	MA	WA	0	1	
Sequential in Keras	Nadam	0.81	0.81	0.79	0.79	0.79	0.79	0.83	0.83	79
VGG19	AdaGrad	0.84	0.84	0.82	0.82	0.82	0.82	0.93	0.93	82
InceptionV3	Nadam	0.81	0.81	0.81	0.81	0.81	0.81	0.88	0.88	81
Inception ResNetV2	Nadam	0.81	0.81	0.81	0.81	0.81	0.81	0.91	0.91	81
BUSI dataset for training and validation (647 images) and UBIBC dataset for testing (100 images)										

on the combined test dataset. The higher accuracy and evaluation metrics, as presented in Figure 8, support the notion that merging two distinct datasets into a hybrid dataset for all stages is superior to using them separately at different stages.

3.7. Experiment 7: Models using a large dataset (UBIBC) for training and validation and a small dataset (BUSI) for testing

This experiment involved training/validating models on a large dataset from UBIBC and testing their performance against smaller datasets from BUSI. The models based on the Sequential and VGG19 architectures fine-tuned by TL demonstrated better performance with metrics precision (0.93), recall (0.92), F1-score (0.92), and accuracy (92%), underscoring their ability to make positive predictions and effectively capture positive instances. AUC values of 0.99 reflect excellent discriminatory power, particularly in benign and malignant cases, as shown in Figure 9.

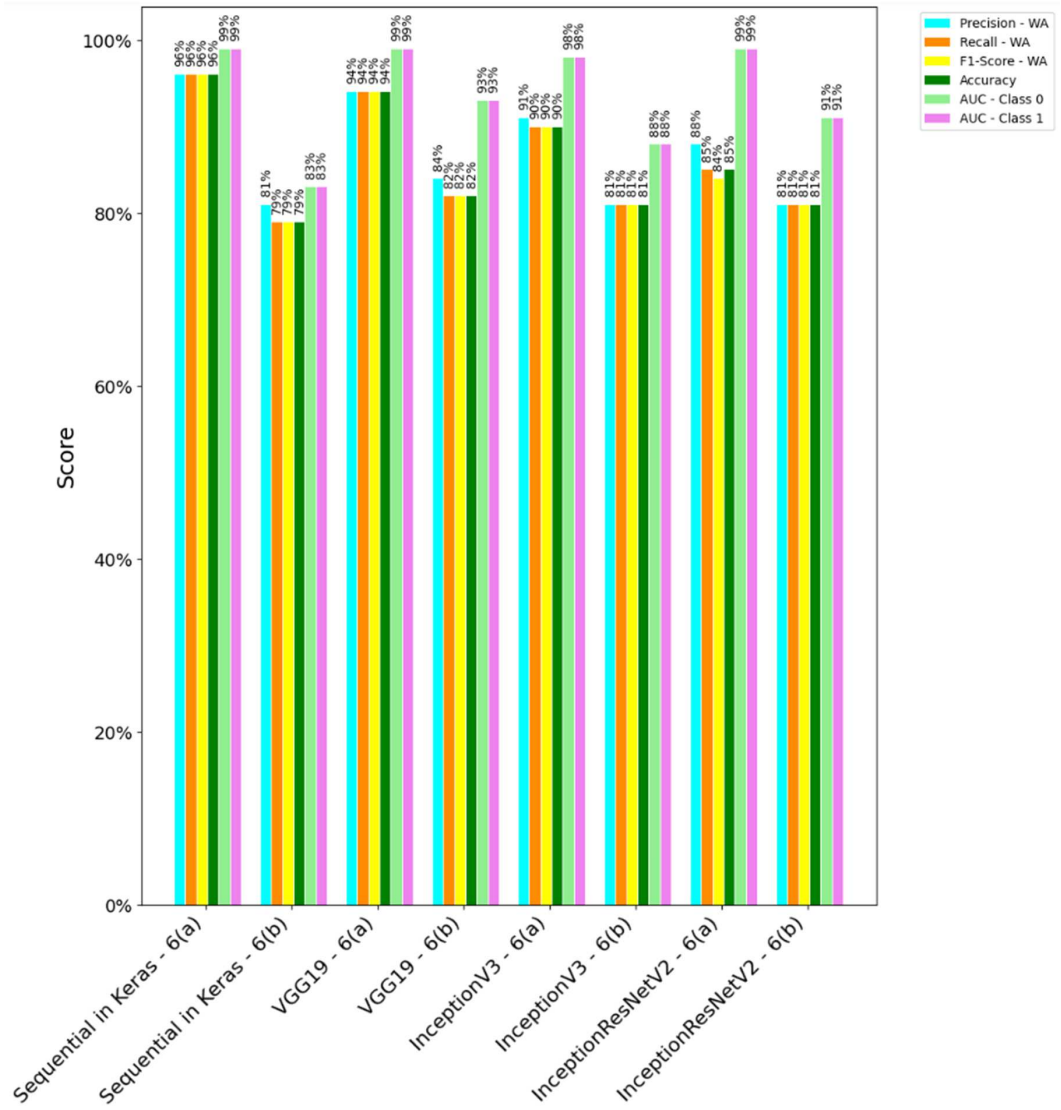
Figure 10 demonstrates that while both models generalized well when trained on the large UBIBC dataset and tested on the smaller BUSI dataset, the sequential model outperformed VGG19. It achieved superior accuracy, balanced precision–recall performance, and slightly better classification consistency, highlighting its robustness in cross-dataset evaluation.

4. Experimental Result Arguments

This study extends the understanding of stepwise implementation techniques for image classification and their significant impact on model performance. Different alternative experiments were implemented and analyzed to explore an efficient way of developing an image classifier. The results of experiment 1 highlighted that models with TL generally outperform those without TL, demonstrating the effectiveness of leveraging pre-trained weights from large datasets like ImageNet. This finding has also been aligned with the existing literature [3–5, 10–13, 15, 16, 18, 21, 22, 25], reinforcing the value of TL in enhancing feature extraction and generalization, especially in medical imaging, where a large volume of data is not often available. The results of experiment 2 further emphasized the importance of DA techniques in improving model robustness and performance. However, there is a limitation that some combinations of DA techniques are better than others, so applying them blindly without considering the nuances of a particular problem/dataset may generate lower accuracy models. However, the combination of horizontal and vertical flips, shifts, shear, rotation, zoom, and cropping proved particularly effective in this study. These techniques have been validated in other studies [4, 5, 8, 10–14, 16, 23–25] as well, showing their efficacy in creating diverse training samples that enhance model generalization.

Furthermore, the exploration of hyperparameter tuning has also been implemented in experiment 3. The results were indicative of the nuanced effects of different optimizers (AdaGrad, SGD, Nadam) and configurations, such as learning rates, dropout rates, batch sizes, convolution blocks, batch normalization, early stopping, L2 regularization, and activation functions on model performance. This result underscored the necessity of tailored hyperparameter tuning for achieving optimal performance in DL models. For example, different CNN architectures may require different optimizers [16, 21] for optimal performance, and they may require different numbers of frozen layers during fine-tuning [3, 5]. Going further, stacked ensemble models from Experiment 4 did not yield significant performance improvements, highlighting the challenges of working with relatively small datasets. This finding is aligned with the broader understanding that

Figure 8
Comparing the metrics of experiments 6(a) and 6(b)



ensemble methods often require large amounts of data to realize their full potential [26]. The experiment underscored the importance of dataset size and diversity in achieving substantial benefits from ensemble techniques.

Experiments 5 and 6 focused on the impact of dataset fusion and image clarity on model performance. The improvements observed in models using combined datasets and the superior performance of models on the clearer dataset underscore the critical role of data quality and diversity, suggesting that mixing high-quality images could be a useful procedure in general. The results of experiment 5 highlighted the fact that optimizing DL models sometimes involves trade-offs with the achievement of a higher accuracy at the cost of a lower AUC score. Experiment 6, particularly the comparison between combined datasets for training, validation, and testing (BUSI + UBIBC, split into training, validation, and testing), led to the highest performance, with Sequential achieving 96% accuracy and VGG19 94% accuracy. This demonstrates that combining datasets enhances sample diversity, improves robustness, and reduces overfitting. Cross-dataset experiments 6a and 7, where models were trained on one dataset and tested on another, highlighted the role of dataset size in generalizability. Training on the smaller BUSI dataset and testing on the larger UBIBC resulted in lower performance (Sequential 79%, VGG19 82% accuracy), reflecting limited transferability from small to large, diverse datasets. It means that training on a small dataset limits the model's ability to generalize to larger, more varied datasets. Conversely, experiment 7 demonstrated the robustness of models trained on a large dataset and tested with different smaller datasets (BUSI, yielding higher performance (Sequential 92%, VGG19 90% accuracy), indicating that models trained on larger datasets generalize well to smaller external datasets. This experiment underscores the potential of large, high-quality datasets in training robust models capable of generalizing well to different datasets.

In summary, following the step-by-step approach not only standardizes the image classification tasks but also achieves better results than comparable studies, such as the work presented by Balaha et al. [21], which employed the fusion of BUSI and BUI datasets to assess diverse DL architectures: InceptionResNetV2, ResNet152, VGG19, and Xception, each utilizing a specific

Figure 9
Metrics for UBIBC (training and validation) and BUSI (testing)

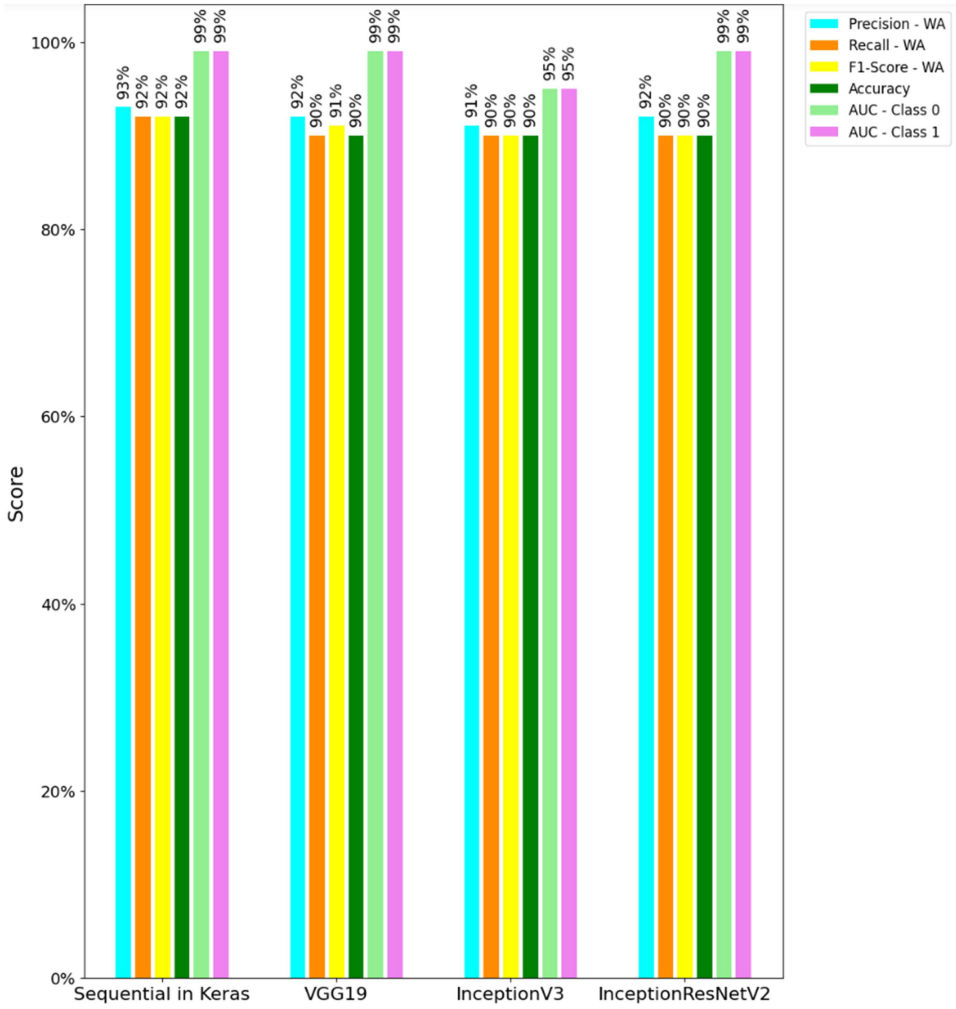
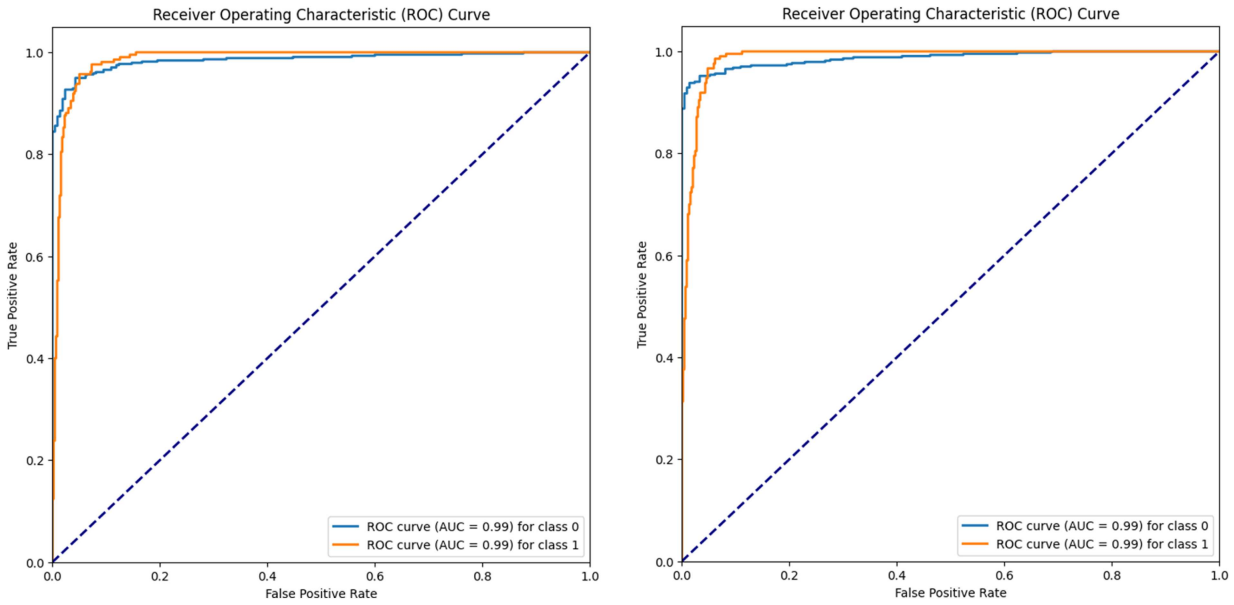


Figure 10
AUC-ROC curve of VGG19 vs sequential



optimizer. Their study revealed distinct performance outcomes, with InceptionResNetV2 achieving the highest accuracy (88.45%), precision (89.80%), recall (87.04%), F1-score (88.38%), and AUC at 97.04%. In contrast, this research comes across the sequential model's superiority over other models, as the sequential model achieved accuracy (92%), precision (93%), recall (92%), F1-score (92%), and AUC (99%), demonstrating superior accuracy. This model had robust classification, minimized false positives, and shorter training time, emphasizing its suitability for medical classification tasks. While the model demonstrates strong performance across multiple datasets, several limitations were also observed, particularly in terms of dataset selection and generalizability. The model's performance varied when trained on one dataset and tested on another, highlighting the challenge of dataset generalization. This suggests that while the model performs well on the specific datasets used in this study, its effectiveness may decrease when applied to different datasets without further fine-tuning. This limitation could impact the generalizability of the findings to other applications or imaging modalities, such as mammography or MRI, where data characteristics differ significantly from ultrasound.

5. Discussion and Conclusions

The use of CNNs is increasingly common, and this DL architecture is an essential component of image classification. Major challenges in building CNN-based image classification applications are the availability of wider architectures, hyperparameter tuning targets and techniques, and DA techniques. A common approach can help to minimize redundancy, facilitate model comparison, streamline communication, encourage quality control, and promote upscaling. This paper proposes a step-by-step approach for building image classifiers and exemplifies it by reporting how to build CNN models to classify ultrasound breast cancer images within the framework. It underscores the critical roles of TL, DA, hyperparameter tuning, and dataset fusion in achieving robust and high-performing models. The step-by-step approach is a checklist to execute combinations relevant to the CNN-based applications with the image classification task. Besides, the wider range of experimentation, analyses, and discussions contributes to the broader understanding of DL applications in imaging and offers valuable guidance for future research and applications. Through an extensive exploration of DL architecture, TL, and DA techniques, it has been concluded that incorporating a diverse range of training datasets is crucial for achieving optimal model performance. Specifically, using varied datasets, including those with different image qualities and characteristics, significantly enhanced the model's ability to generalize and perform accurately across various scenarios. However, some combinations of DA techniques outperform others, so applying them without accounting for the specific characteristics of the problem or dataset can lead to reduced model accuracy. The systematic methodology, encompassing TL, DA, hyperparameter tuning, ensemble models, and data diversity, provides both a methodology for and valuable insights into the common challenge of optimizing DL models for image classification.

This approach can be a roadmap for future endeavors in improving the fast-growing field of image-based computational medicine. More generally, we hope that the approach will contribute to a standardized development of image classification applications that align with Jerome Bruner's theory of scaffolded learning [27], which states that when learners are exposed to medical image classification for the first time, they need active support. By navigating them with the aid of a structured approach, anyone building a CNN-based image classifier gains confidence and independence by following Kolb's model of experiential learning [28]. For example, if students are new to image classification, the presented approach would help them retain and recall aspects of their first experience of building and training a CNN. While reflecting, they could compare their experience with each step in the approach to identify particularly challenging or novel ideas. Finally, if they had the chance to revisit the problem by following the approach again, they could improve on their previous approach by applying their new ideas. Overall, this step-by-step approach would act like the scaffolding that supports a building under construction, which is gradually taken down as the building gains more parts. In addition, the step-by-step approach deploys one of the nine ways (segmentation) to reduce cognitive load recommended by [29]. By breaking the complex task of building CNN models for medical image classification into small segments or well-defined steps, the approach could potentially facilitate learning, especially for neurodivergent students [30].

Another specific advantage relates to classroom discussion, as argued by Hollander [31]. The proposed approach can be used as a tool for the discussion, where different students read about different steps in the approach and hence structure the discussion based on the approach, and minimize digression by asking the students to adhere to the well-defined topics. Moreover, it can also be useful in practical group assignments where an instructor could assign different students to different parts of the approach before asking them to work together to build a CNN-based image classifier. By creating interdependence and judging the group collectively, the instructor would be creating an environment conducive to the jigsaw method of cooperative learning [32]. The instructor could even create multiple groups in this way to promote knowledge diffusion, encouraging this syndicate of groups to devise diverse solutions (multiple scaffolds) to the same problem [33]. In this hypothetical environment, a student would have to complete some tasks independently and creatively (out-of-the-box problem-solving [34]) while benefiting from multiple chances to improve on their limited solution through their peers' perspectives.

Over and above that, one specific benefit of the presented approach is the combinatorial explosion, which is a major challenge in building CNN-based AI applications. There is a wide range of architectures, hyperparameter tuning targets and techniques, and DA techniques. The presented approach is like a checklist to help researchers exhaust the combinations relevant to a medical image classification task. Finally, we have confidence that the organized step-by-step approach will help to classify medical images, which is increasingly common, and this DL architecture is an essential component of image classification–based applications. Moreover, this approach brings a common approach that will minimize redundancy, facilitate model comparison, streamline communication, encourage quality control, and promote upscaling. It could also serve as a pedagogical tool. For example, an instructor could base their rubric and grading criteria on our approach. They could use the approach to streamline group work and problem-based learning, too.

To explore further, future research should focus on a systematic evaluation of TL strategies, DA combinations, and hyperparameter optimization techniques across larger, multi-institutional datasets encompassing varied imaging modalities (e.g., ultrasound, mammography, MRI). Such investigations would provide deeper insights into the generalizability and robustness of CNN-based models in clinical contexts. Furthermore,

extending the proposed step-by-step framework into automated pipelines, incorporating methods such as Bayesian optimization, AutoML, or Neural Architecture Search, could reduce manual trial and error, enhance reproducibility, and accelerate the deployment of high-performing models in medical image classification.

Ethical Statement

The authors declare that this study did not require formal ethical approval because it involved a secondary analysis of publicly available medical imaging datasets (e.g., BUSI and UBIBC) for training and performance evaluation of machine learning models (CNNs), without any direct contact with or intervention involving human subjects.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

The data that support the findings of this study are openly available in the BUSI at https://doi.org/10.1016/j.dib.2019.104863, in BUI data at https://doi.org/10.17632/wmy84gzngw.1, and UBIBC in Kaggle at https://www.kaggle.com/datasets/vuppalaadithyasairam/ultrasound-breast-images-for-breast-cancer/.

Author Contribution Statement

Ahamadullah Lohani: Conceptualization, Methodology, Software, Formal analysis, Investigation, Data curation, Writing – original draft, Writing – review & editing, Visualization. **Bhupesh Kumar Mishra:** Conceptualization, Methodology, Validation, Investigation, Writing – review & editing, Supervision, Project administration. **Kenneth Y. Wertheim:** Validation, Investigation, Writing – review & editing. **Temitayo Matthew Fagbola:** Validation, Investigation, Writing – review & editing.

References

[1] Sarvamangala, D. R., & Kulkarni, R. V. (2022). Convolutional neural networks in medical image understanding: A survey. *Evolutionary Intelligence*, *15*(1), 1–22. https://doi.org/10.1007/s12065-020-00540-3

[2] Salehi, A. W., Khan, S., Gupta, G., Alabduallah, B. I., Almjally, A., Alsolai, H., . . . , & Mellit, A. (2023). A study of CNN and transfer learning in medical imaging: Advantages, challenges, future scope. *Sustainability*, *15*(7), 5930. https://doi.org/10.3390/su15075930

[3] Kim, H. E., Cosa-Linan, A., Santhanam, N., Jannesari, M., Maros, M. E., & Ganslandt, T. (2022). Transfer learning for medical image classification: A literature review. *BMC Medical Imaging*, *22*(1), 69. https://doi.org/10.1186/s12880-022-00793-7

[4] Arooj, S., Zubair, M., Khan, M. F., Alissa, K., Khan, M. A., & Mosavi, A. (2022). Breast cancer detection and classification empowered with transfer learning. *Frontiers in Public Health*, *10*, 924432. https://doi.org/10.3389/fpubh.2022.924432

[5] Korzhebin, T. A., & Egorov, A. D. (2021). Comparison of combinations of data augmentation methods and transfer learning strategies in image classification used in convolution deep neural networks. In *2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering*. 479-482.https://doi.org/10.1109/ElConRus51938.2021.9396724

[6] Barati, B., Erfaninejad, M., & Khanbabaei, H. (2025). Evaluation of effect of optimizers and loss functions on prediction accuracy of brain tumor type using a Light neural network. *Biomedical Signal Processing and Control*, *103*, 107409. https://doi.org/10.1016/j.bspc.2024.107409

[7] Nanni, L., Paci, M., Brahnam, S., & Lumini, A. (2021). Comparison of different image data augmentation approaches. *Journal of Imaging*, *7*(12), 254. https://doi.org/10.3390/jimaging7120254

[8] Khosla, C., & Saini, B. S. (2020). Enhancing performance of deep learning models with different data augmentation techniques: A survey. In *2020 International Conference on Intelligent Engineering and Management*, 79–85. https://doi.org/10.1109/ICIEM48762.2020.9160048

[9] Mohammed, F. A., Tune, K. K., Assefa, B. G., Jett, M., & Muhie, S. (2024). Medical image classifications using convolutional neural networks: A survey of current methods and statistical modeling of the literature. *Machine Learning and Knowledge Extraction*, *6*(1), 699–735. https://doi.org/10.3390/make6010033

[10] Pacal, İ. (2022). Deep learning approaches for classification of breast cancer in ultrasound (US) images. *Journal of the Institute of Science and Technology*, *12*(4), 1917–1927. https://doi.org/10.21597/jist.1183679

[11] Reenadevi, R., Sathiya, T., & Sathiyabhama, B. (2021). Breast cancer histopathological image classification using augmentation based on optimized deep ResNet-152 structure. *Annals of the Romanian Society for Cell Biology*, *25*(6), 5866–5874.

[12] Salama, W. M., & Aly, M. H. (2021). Deep learning in mammography images segmentation and classification: Automated CNN approach. *Alexandria Engineering Journal*, *60*(5), 4701–4709. https://doi.org/10.1016/j.aej.2021.03.048

[13] Al-Dhabyani, W., Gomaa, M., Khaled, H., & Aly, F. (2019). Deep learning approaches for data augmentation and classification of breast masses using ultrasound images. *International Journal of Advanced Computer Science and Applications*, *10*(5), 1–11. https://doi.org/10.14569/IJACSA.2019.0100579

[14] Saber, A., Sakr, M., Abo-Seida, O. M., Keshk, A., & Chen, H. (2021). A novel deep-learning model for automatic detection and classification of breast cancer using the transfer-learning technique. *IEEE Access*, *9*, 71194–71209. https://doi.org/10.1109/ACCESS.2021.3079204

[15] Kumar, S. J. K., Parthasarathi, P., Hogo, M. A., Masud, M., Al-Amri, J. F., & Abouhawwash, M. (2023). Breast cancer detection using breastnet-18 augmentation with fine tuned VGG-16s breast cancer detection using breastnet-18 augmentation with fine tuned VGG-16. *Intelligent Automation & Soft Computing*, *36*(2), 2363–2378. https://doi.org/10.32604/iasc.2023.033800

[16] Castro-Tapia, S., Castaneda-Miranda, C. L., Olvera-Olvera, C. A., Guerrero-Osuna, H. A., Ortiz-Rodriguez, J. M., Martinez-Blanco, M. D. R., . . . , & Solis-Sanchez, L. O. (2021). Classification of breast cancer in mammograms with deep learning adding a fifth class. *Applied Sciences*, *11*(23), 11398. https://doi.org/10.3390/app112311398

[17] Wakili, M. A., Shehu, H. A., Sharif, M. H., Sharif, M. H. U., Umar, A., Kusetogullari, H., . . . , & Uyaver, S. (2022). Classification of breast cancer histopathological images using DenseNet and transfer learning. *Computational*

Intelligence and Neuroscience, 2022(1), 8904768. https://doi.org/10.1155/2022/8904768

[18] Ayana, G., Park, J., Jeong, J. W., & Choe, S. W. (2022). A novel multistage transfer learning for ultrasound breast cancer image classification. *Diagnostics*, *12*(1), 135. https://doi.org/10.3390/diagnostics12010135

[19] Yang, S., Zhu, F., Ling, X., Liu, Q., & Zhao, P. (2021). Intelligent health care: Applications of deep learning in computational medicine. *Frontiers in Genetics*, *12*, 607471. https://doi.org/10.3389/fgene.2021.607471

[20] Cui, M., & Zhang, D. Y. (2021). Artificial intelligence and computational pathology. *Laboratory Investigation*, *101*(4), 412–422. https://doi.org/10.1038/s41374-020-00514-0

[21] Balaha, H. M., Saif, M., Tamer, A., & Abdelhay, E. H. (2022). Hybrid deep learning and genetic algorithms approach (HMB-DLGAHA) for the early ultrasound diagnoses of breast cancer. *Neural Computing and Applications*, *34*(11), 8671–8695. https://doi.org/10.1007/s00521-021-06851-5

[22] Kathamuthu, N. D., Subramaniam, S., Le, Q. H., Muthusamy, S., Panchal, H., Sundararajan, S. C. M., . . . , & Zahra, M. M. A. (2023). A deep transfer learning-based convolution neural network model for COVID-19 detection using computed tomography scan images for medical applications. *Advances in Engineering Software*, *175*, 103317. https://doi.org/10.1016/j.advengsoft.2022.103317

[23] Abdou, M. A. (2022). Literature review: Efficient deep neural networks techniques for medical image analysis. *Neural Computing and Applications*, *34*(8), 5791–5812. https://doi.org/10.1007/s00521-022-06960-9

[24] Yang, S., Xiao, W., Zhang, M., Guo, S., Zhao, J., & Shen, F. (2022). Image data augmentation for deep learning: A survey. *arXiv Preprint: 2204.08610*

[25] Ayana, G., Dese, K., & Choe, S. W. (2021). Transfer learning in breast cancer diagnoses via ultrasound imaging. *Cancers*, *13*(4), 738. https://doi.org/10.3390/cancers13040738

[26] Wu, H., & Levinson, D. (2021). The ensemble approach to forecasting: A review and synthesis. *Transportation Research Part C: Emerging Technologies*, *132*, 103357. https://doi.org/10.1016/j.trc.2021.103357

[27] Wood, D., Bruner, J. S., & Ross, G. (1976). The role of tutoring in problem solving. *Journal of Child Psychology and Psychiatry*, *17*(2), 89–100. https://doi.org/10.1111/j.1469-7610.1976.tb00381.x

[28] Sugarman, L. (1985). Kolb's model of experiential learning: Touchstone for trainers, students, counselors, and clients. *Journal of Counseling & Development*, *64*(4), 264–268. https://doi.org/10.1002/j.1556-6676.1985.tb01097.x

[29] Mayer, R. E., & Moreno, R. (2003). Nine ways to reduce cognitive load in multimedia learning. *Educational Psychologist*, *38*(1), 43–52. https://doi.org/10.1207/S15326985EP3801_6

[30] Ganju, G., Kumar, A., & Uma, M. (2024). Design and development of cognitive skills enhancement for neurodivergent children using CNN. In *2024 International Conference on Advances in Computing, Communication and Applied Informatics*, 1–5. https://doi.org/10.1109/ACCAI61061.2024.10601914

[31] Hollander, J. A. (2002). Learning to discuss: Strategies for improving the quality of class discussion. *Teaching Sociology*, *30*(3), 317–327. https://doi.org/10.2307/3211480

[32] Jeppu, A. K., Kumar, K. A., & Sethi, A. (2023). 'We work together as a group': Implications of jigsaw cooperative learning. *BMC Medical Education*, *23*(1), 734. https://doi.org/10.1186/s12909-023-04734-y

[33] Haynes, A., Haynes, K., Habeshaw, S., Gibbs, G., & Habeshaw, T. (2020). *53 interesting things to do in your lectures: Tips and strategies for really effective lectures and presentations*. London: Routledge. https://doi.org/10.4324/9781003114741

[34] Walker, A., Leary, H., & Hmelo-Silver, C. (2015). *Essential readings in problem-based learning: Exploring and extending the legacy of Howard S. Barrows*. USA: Purdue University Press.

How to Cite: Lohani, A., Mishra, B. K., Wertheim, K. Y., & Fagbola, T. M. (2026). Step-by-Step Approach to Design Image Classifiers in AI: An Exemplary Application of the CNN Architecture for Breast Cancer Diagnosis. *Artificial Intelligence and Applications*. https://doi.org/10.47852/bonviewAIA52025938