Deep Transfer Learning for Breast Cancer Classification

P. Djagba and J. K. Buwa Mbouobda*

Department of mathematics, Nelson Mandela University, South Africa

African Institute for Mathematical Sciences, Ghana

DOI: https://doi.org/10.51244/IJRSI.2025.12010023

Received: 23 December 2024; Accepted: 30 December 2024; Published: 03 February 2025

INTRODUCTION

According to GLOBOCAN 2020, there were 10.3 million cancer-related deaths and 19.3 million new cases of the disease worldwide. Female breast cancer represented 11.7% of all cases, or 2.26 million cases, globally [10, 31]. In Africa, year 2020, 85 787 women died from breast cancer out of 186598 new cases, 12.5% of world's death by breast cancer [31]. In the same year, we diagnosed 2262419 new cases of breast cancer in the world [31]. Through the use of computer-aided procedures, the accuracy of diagnosis may be significantly boosted. One of the most efficient methods to detect breast cancer is a pathologist's examination of histopathological pictures under a microscope [9, 12]. Since the invention of digital image scanners, picture identification has improved [3]. The practice of digital pathology has advanced significantly, in part because whole slide image (WSI) scanners enable quick and affordable diagnosis [8]. Breast cancer of the Invasive Ductal Carcinoma (IDC) kind comes from a dysfunction of milk duct's cells of the breast and eventually spreads to the breast tissue nearby. It is an example of cancer type where early detection and accurate classification can significantly improve patient outcomes. The results for patients can be considerably improved by IDC early identification and precise categorization. On the basis of histopathological scans, IDCs have been accurately classified using machine learning approaches [1, 26]. Diagnosing pathological imaging is difficult and necessitates manual skills and a microscope. Due to the limitations of human cognition, this time-consuming approach frequently results in mistakes. One might anticipate time savings and decreased mistake rates with an automated system [26]. One of the main challenges in analyzing cancer images using deep learning algorithms is the limited availability of data [23, 29]. Deep learning algorithms require large amounts of data to train effectively, and the lack of data can lead to overfitting or underfitting of the model [23]. Many models have been created to detect malignant cells for quite some time, and deep learning models have been used in that regard. [24]. But, one can ask that: can deep learning algorithms accurately predict the presence of cancer tumor in image data? In this paper, we are going to use Vision Transformers, VGG, and ResNets to classify images of breast cancer.

METHODOLOGY

This section describes the methodology employed to analyze the cancer dataset. It concentrates on convolutional neural network (CNN) architectures like ResNet, VGG in addition to the emerging field of vision transformers. This chapter provides an overview of the dataset, discusses the preprocessing steps, and introduces the selected CNN architectures and their potential cancer classification applications. In addition, the use of vision transformers as an alternative method is highlighted.

Dataset description

Data source

We use the dataset provided at https://bit.ly/3XBOdum.

The majority of occurrences of breast cancer are caused by invasive ductal carcinoma (IDC). When evaluating a specimen's aggressiveness as it is put on the test slide of the microscope, pathologists pay special attention to the areas that contain the IDC. Delineating the specific IDC zones inside a whole-mount

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue I January 2025

microscope slide is therefore a typical preprocessing step for automated aggressiveness evaluation. The original dataset consisted of 162 whole-mount slide images of Breast Cancer (BCa), from women patients at the Hospital of the University of Pennsylvania and The Cancer Institute of New Jersey [1]. The original microscope slides used to create these photos of breast cancer (BCa) were magnified at a level of 40. They are separated into 277,524 patches, each 50 pixels by 50 pixels in size. The IDC-positive class sample count in this total is 78,786; the IDC-negative class sample count is 198,738. Figure 1 presents the dataset's sample pictures.

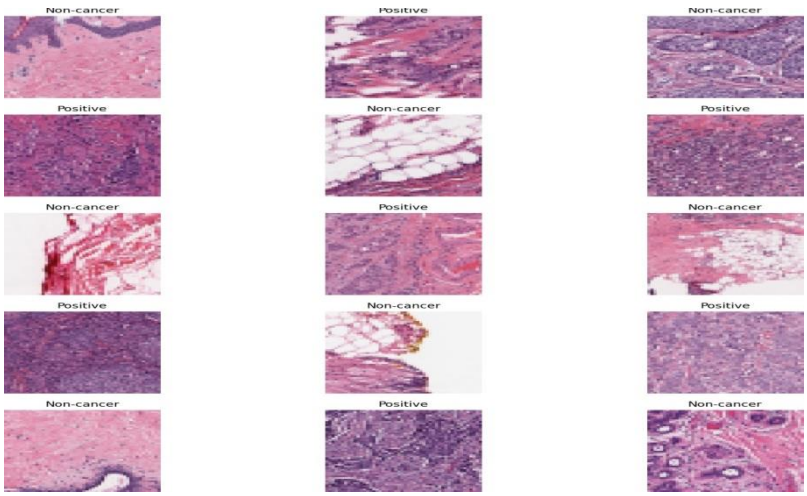

Figure 1: Some examples of images in the breast histopathology imaging dataset for both classes.

Image preprocessing

Both the identification of breast cancer and the accuracy with which a model makes predictions might be hampered by the presence of artifacts and noise. As a result, it is essential to process the images in order to enhance the performance of a model. The aim of pre-processing is an improvement of the image data that suppresses unwilling distortions or enhances some image features important for further processing, although geometric transformations of images (e.g. rotation, scaling, translation) are classified among pre-processing methods here since similar techniques are used [30]. Image augmentation is a method that is used in machine learning and computer vision to artificially extend a dataset by creating new modified versions of images. It is especially helpful when the original dataset has a limited number of samples since it helps minimize overfitting and enhances the performance of deep neural networks.

Here are some important details about the enhancement of images:

Aim: The most important aim of picture augmentation is to enhance the diversity of the dataset. This will enable the model to learn from a greater variety of variations and will improve its capacity to generalize to new data.

Techniques: Image augmentation entails making numerous modifications to the source photos, such as rotating, scaling, cropping, flipping, and adjusting the brightness, among other things. It is possible to build enhanced pictures by combining these alterations or applying them in a random order.

Implementation: Image augmentation can be accomplished via the use of libraries or frameworks that offer built-in functions or application programming interfaces (APIs) for the purpose of creating augmented images. For the purpose of doing picture augmentation, for instance, the Image Data Generator class is often used in deep learning frameworks such as Keras. In this work, we use enhancement methods from Pytorch and Keras frameworks.

Benefit: Image augmentation helps to expand the size of the training dataset, which may lead to improved model performance and generalization. Benefits Image augmentation helps to increase the size of the training dataset, which can lead to improvements in model performance and generalization. In addition to

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue I January 2025

this, it helps to avoid overfitting by injecting variances in the data, which in turn makes the model more resistant to a variety of settings and scenarios.

Picture augmentation is a strong approach that allows for the growth and diversity of picture datasets. This improves the performance and generalization capacity of machine learning models, especially in computer vision applications. In general, image augmentation is a technique that allows for the expansion and diversification of image datasets.

Dataset splitting

It is standard procedure to separate the dataset into three separate sets when working with machine learning models [28]. These sets are referred to as the training set, the testing set, and the validation set respectively. The following is a description of each set, along with the benefits that come with them:

The **training set** is the biggest part of the dataset and is what is utilized to train the machine learning model. It is the set that the model uses to discover the underlying connections and patterns in the data that it is working with. A training set offers the following benefits to its users:

Model learning: The training set enables the model to learn from the data and alter its parameters in order to reduce the amount of mistake or loss function.

Pattern identification: The model is able to discover and capture the underlying patterns and connections in the data if it is trained on a large and varied training set, which is required for the process.

Parameter optimization: The training set is used in the process of optimizing the model's parameters by using methods such as gradient descent or backpropagation.

Testing set: The purpose of the testing set is to assess the performance of the trained model on data that it has not seen before. It is helpful in determining how effectively the model generalizes to new cases that have not been observed before. The following is a list of the benefits of employing a testing set:

Performance evaluation: The testing set provides an objective assessment of the performance of the model when applied to fresh data. It is useful in determining the extent to which the model can generalize and provide reliable predictions.

Overfitting detection: By analyzing the model using the testing set, we may deter-mine whether or not the model has overfit the training data. When a model performs well on the data it was trained on but badly on fresh data, this is an example of overfitting.

Model comparison: The testing set enables the comparison of several models or algorithms in order to determine which one performs the best.

Validation set: The validation set is used in the process of adjusting hyperparameters and selecting models. It does this by fine-tuning the hyperparameters and choosing the model that performs the best overall. This enhances the overall performance of the model. Using a validation set comes with a number of benefits, including the following:

Hyperparameter Tuning: The validation set is used to analyze several possible combinations of hyperparameters in order to pick the model that performs the best. It contributes to the optimization of the model's performance as well as its capacity for generalization.

Preventing Overfitting: If we use a separate validation set, we can avoid overfitting to the testing set and guarantee that the performance of the model is not skewed toward a particular set of hyperparameters. This is possible because we can prevent overfitting to the testing set.

Model Selection: The validation set enables the comparison of many models or architectures in order to

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue I January 2025

pick the one that delivers the highest level of performance.

In a nutshell, one of the most important steps in machine learning is to segment the dataset into training, testing, and validation sets. The validation set is used for hyperparameter tweaking and model selection, while the training set is used to train the model. The testing set is used to assess how well the model performs on data that it has not seen before. This strategy helps ensuring that the model works well on fresh data, prevents overfitting, and enables the selection of the best possible model to use.

Convolutional neural networks

Convolutional Neural Network (CNN) is a deep network that replicates how the human visual cortex processes and recognizes pictures [25]. The Figure 2 below is a graphical representation of a CNN. From that figure (Figure 2), CNN's architecture is made up of neural layers that aid in feature extraction and additional neural layers that serve as classifiers. Convolution layer, pooling layer, activation function, and fully connected layer are the minimum number of components that make up the CNN model [20]. The feature extraction part consists of convolutional and pooling layers, while the fully connected part is the classifier. A feature map is the output picture of a convolutional layer [25]. The convolutional layer (CL) is the main part of a CNN [22]. In the CL, we use a filter to extract features (or patterns) in the input image. The input image is defined as a three-dimensional matrix height*width*depth. The shape height*width is the pixel size of the image and the *depth* represents the number of channel (3 for colored pictures (RGB) and 2 for gray-scale images). The input image is then convoluted with the kernel (filter), which results to the feature map. The convolution equation for each input image is given by [22]:

$$Feature\_map_j = \sum_{i=1}^{N} I_i * K_{i,j} + B_j$$

where the $\mathbf{I}_j$ represent the input matrices, $\mathbf{K}_{i,j}$ the corresponding kernels and $\mathbf{B}_j$, the biases. The pooling layer helps to reduce the dimension of the input. It chooses the most relevant features from the feature map and passes the result to the activation layer.

The activation layer introduces the non-linearity in the model. They are many activation functions. Some of them are: the sigmoid function, the softmax function, the hyperbolic tangent function and the Rectified Linear Unit (ReLU) function.

The equations are given below.

(sigmoid function) $\sigma(x) = \frac{1}{1+e^{-x}}$

(Hyperbolic tangent) $\tanh(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$

(ReLU) $ReLU(x) = \max(0,x) = \begin{cases} x \; if \; x > 0; \\ 0 \; if \; x \leq 0. \end{cases}$

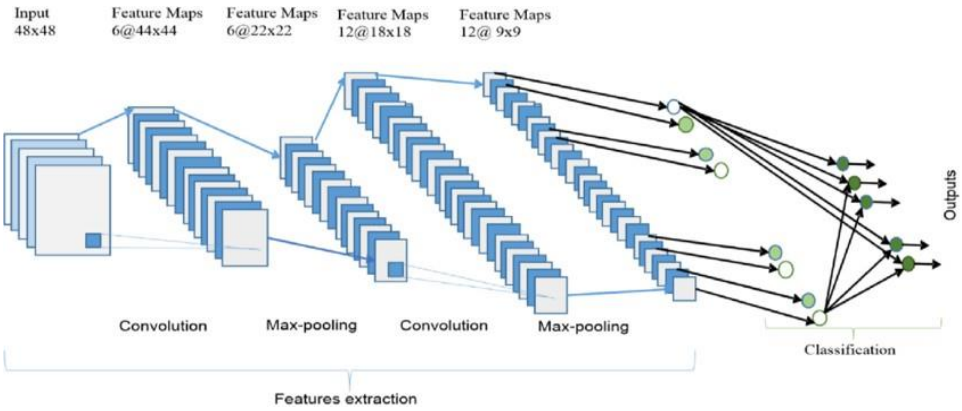

Figure 2: Typical architecture of a convolutional neural network

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue I January 2025

(Soft max) $\sigma(Z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$ for $i = 1, 2, 3, \dots, K$ and $Z = (z_1, z_2, \dots, z_K) \in \mathbb{R}^K$.

Recent studies pointed the ReLU function to be a good activation function for classification problems using CNN as its improves both the learning speed and the performance of the models [21]. In this work we will use a ReLU activation function. The next part is a set of fully connected layers (a basic neural network) which helps for classification.

The VGG architecture

Deepchecks [14] describes the VGG architecture as a convolutional neural network (CNN) architecture that has been in use for some time. It was created by the Visual Geometry Group (VGG) at the University of Oxford as a consequence of research on how to make certain networks denser [7]. The network is distinguished by its simplicity, with the only additional components being pooling layers and a completely linked layer [11]. The network makes use of modest filters measuring 3 by 3, and it is also known for its compact size. The architecture is a conventional deep CNN architecture with numerous layers, and the term deep refers to the number of layers with VGG-16 or VGG-19 consisting of 16 or 19 convolutional layers, respectively [7, 13]. The network receives an image with the dimensions (224, 224, 3) as its input [18]. The first two layers have the same padding and each feature 64 channels with a filter size of 3 by 3. Following that, a max pool layer with a stride of (2,2) is followed by two convolution layers with a filter size of 128 and (3,3). After this comes a max-pooling layer with the stride (2,2), which is the same as the layer before it. After that, there are two convolution layers with a filter size of (3, 3) and 256 filters [18]. One of the most widely used designs for picture recognition is called the VGG architecture, and it serves as the foundation for models of object identification that have broken new ground [7]. In addition, VGG-Net performs better than the baselines on a wide variety of tasks and datasets, not only ImageNet [18]. All of the hidden layers in VGG employ ReLU, which considerably cuts down on the amount of time required for training. On the other hand, Local Response Normalization (LRN) is not used by VGG since it both increases the amount of memory consumed and the amount of time required for training without enhancing accuracy [14]. The architecture which will be used in this essay will be a VGG16. The figure 3 is a diagram of the network.

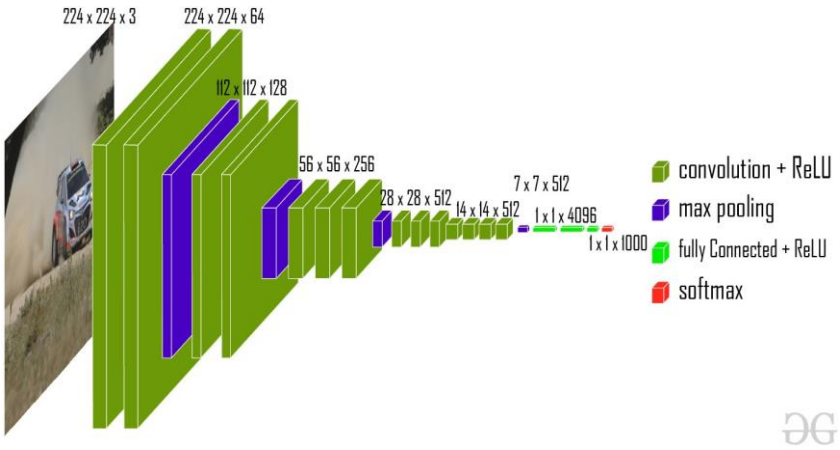

Figure 3: The VGG16 architecture

Residual networks (ResNet)

It has become difficult to talk about CNN architectures without mentioning residual networks. Residual Networks (abbreviated as ResNet) were originated by [19] in 2015 and were afterwards launched by Microsoft Research, winning many awards. The idea came out to solve the problem of vanishing or exploding gradient when it comes to build deep neural networks. Resnet are built with residual connections, forming residual blocks (Figure 4), which allows the information (the learning process) to navigate through the network and prevent gradient from dissipating [6]. The residual connections here refer to a type of link that bypasses one or more layers in the network and reach the output directly. This shortcut connection enables the network to learn fast and to achieve excellent performances.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue I January 2025

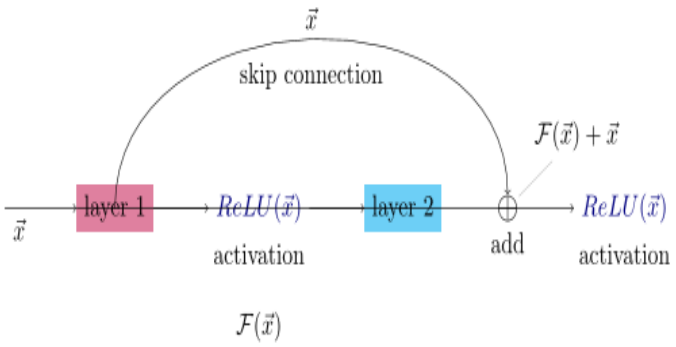

Figure 4: A residual block in the ResNet architecture

There are many ResNet models depending on how the residual blocks are stacked together. The following table present some of the architectures.

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
		3×3 max pool, stride 2				
conv2_x	56×56	[3×3, 64; 3×3, 64] ×2	[3×3, 64; 3×3, 64] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3
conv3_x	28×28	[3×3, 128; 3×3, 128] ×2	[3×3, 128; 3×3, 128] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×8
conv4_x	14×14	[3×3, 256; 3×3, 256] ×2	[3×3, 256; 3×3, 256] ×6	[1×1, 256; 3×3, 256; 1×1, 1024] ×6	[1×1, 256; 3×3, 256; 1×1, 1024] ×23	[1×1, 256; 3×3, 256; 1×1, 1024] ×36
conv5_x	7×7	[3×3, 512; 3×3, 512] ×2	[3×3, 512; 3×3, 512] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Figure 5: Some ResNet architectures

Vision Transformer for image classification

Description

Transformers is a deep learning algorithm that relies on attention [32]. It is known for his high performance in Natural Language Processing tasks. While CNN-based algorithms were seen as the best models for computer vision, vision transformer has shown that if the attention applied in NLP is applied to patches of images, it can perform very well on image classification tasks [15]. Vision transformers, when pretrained on very large image datasets (ImageNet, CIFAR-100, JFT-300M, etc...) and fine-tuned to the target data, performs very well on image recognition tasks compare to ResNet and other CNN models [15].

Figure 6: Example of attention on image input

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue I January 2025

The attention mechanism is a dot product operation between two image features. The vision transformer is made of many blocks. Each of them plays a crucial role in processing the image.

Patches creation

At the very first step of the vision transformer algorithm the image is split in to patches of size P. For example, an image of size 320 × 240, can be split into 300 patches of size 16 × 16. In the Figure 7 below, our images are of size 72 × 72 and we create 144 patches of size 6 × 6.

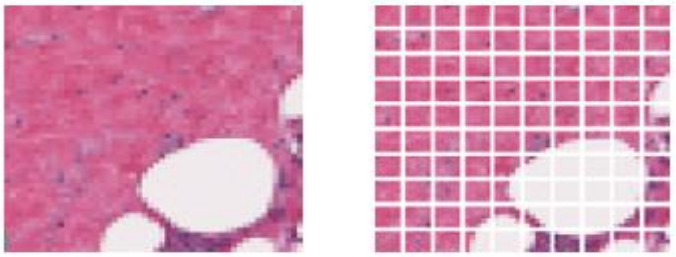

Figure 7: Patches creation

Linear projection of patches

The different patches are now flattened into a vector by projection. The number of patches N is the vector length. Let $x \in \mathbb{R}^{N \times P^2 C}$ (C is the number of channels) an image, a patch can be represented by $x_P^i \in \mathbb{R}^{P \times P \times C}$. We reshape first the patches.

$$x_P^i \in \mathbb{R}^{P \times P \times C} \rightarrow x_P^i \in \mathbb{R}^{1 \times P^2 C}.$$

Next, we compute the projection operation

$$x_P^i \in \mathbb{R}^{1 \times P^2 C} \cdot W \in \mathbb{R}^{P^2 C \times D} \rightarrow x_P^i \cdot W \in \mathbb{R}^{1 \times D}$$

Here, D represent the target dimension of the projection. The operation can be written directly as

$$x \in \mathbb{R}^{N \times P^2 C} \cdot W \in \mathbb{R}^{P^2 C \times D} = x \cdot W \in \mathbb{R}^{N \times D}.$$

The final result is called the patch embedding [15].

Position embedding

The position embeddings are added to the patch embeddings to save the position information of the patches. They are various positions embedding method including sinusoidal function used in the original Transformers paper [32] and Rotary positional embedding (RoPE), well-suited for NLP tasks.

Transformer encoder

This part constitutes the main block of the transformers architecture. It helps to compute the attention. The Transformer encoder [32] is built with alternating layers of multiheaded self-attention and Multi-Layer Perceptron (MLP) blocks. We apply normalize the input before every block, and residual connections after every block [5,33].

The attention is computed using the formula

$$Attention(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = softmax\left(\frac{\boldsymbol{QK^T}}{\sqrt{d_K}}\right)\boldsymbol{V}.$$

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue I January 2025

Where **Q** (query) represent the features of interest, **K** (key) are the relevant features, **V** (value) is the original feature and d_K is a normalization factor.

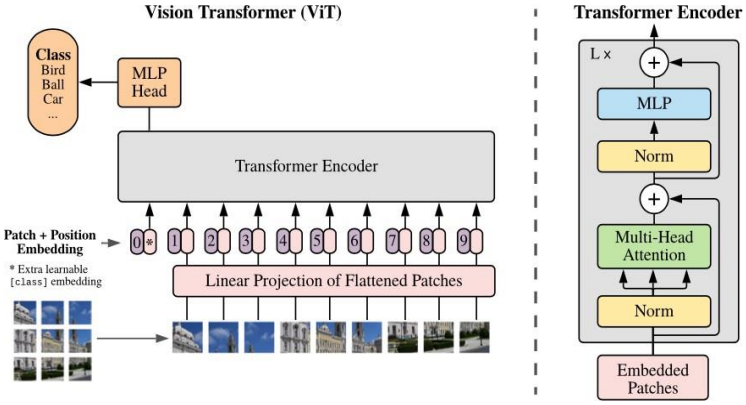

Figure 8: The overall vision transformer architecture

The normalization reduces the training time, and stabilize the training while the skip of connections improves the performance.

Prediction module

The final part of the vision transformer serves to classification. It is made of dense layers like in CNN. It is a set of MLP helping to categorize images.

Model evaluation

When it comes to the evaluation of a model that uses machine learning, there are a variety of methods and metrics that may be employed. We present here some important aspects to consider while evaluating models.

Accuracy

Accuracy is a measurement that determines how right the model's predictions are in general. It is determined by determining the ratio of the number of accurate forecasts to the total number of predictions that were made. On the other hand, accuracy by itself could not be enough in situations in which the dataset is unbalanced or in which the effects of several kinds of mistake vary greatly from one another,

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} = \frac{Correct\ predictions}{All\ prediction\ instances}.$$

Precision

Precision is defined as the ratio of accurately predicted positive occurrences to the total number of cases that were anticipated to be positive. It focuses on the capability of the model to prevent producing false positives. The ratio of the number of true positives to the total number of true positives and false positives is the formula for calculating precision,

$$Precision = \frac{TP}{TP + FP} = \frac{True\ positive}{Total\ positive\ predictions}$$

Recall

Recall, which also goes by the names sensitivity and true positive rate, is a measurement that determines the percentage of properly predicted positive cases out of all actual positive instances. It places an emphasis on the model's capacity to recognize all instances of good outcomes. The ratio of the number of true positives to the total number of true positives and false negatives is how recall is determined,

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue I January 2025

$$Recall = \frac{TP}{TP + FN} = \frac{True\ positive}{Total\ positive\ observations}.$$

The F1-score

TheF1-score is a statistic that combines precision and recall into a single number. This value is referred known as the F1-score. Due to the fact that it takes into consideration both false positives and false negatives, it offers a more accurate measurement of the performance of the model. The F1-score is determined by taking the harmonic mean of the recall and precision scores,

$$F1 - score = 2 \times \frac{Recall \times precision}{Recall + precision}.$$

Confusion matrix

A confusion matrix is a table that describes the performance of a classification model. It presents a comprehensive analysis of the model's predictions, including the model's true positives, true negatives, false positives, and false negatives. It is helpful for comprehending the various kinds of mistakes that the model is producing.

In order to have a thorough comprehension of the performance of the model, it is essential to take into account a variety of assessment metrics and approaches. The particular job, dataset, and objectives of the project all play a role in determining which assessment technique to use. We are able to arrive at well-informed conclusions and increase the model's performance if we give great consideration to its evaluation.

Transfer learning and Fine-tuning

Transfer learning

Transfer learning is the process of applying deep learning models that have been trained on a big dataset to a smaller dataset in order to solve a different issue. This is accomplished by transferring the weights that were learned during a prior training phase to a new training session. The VGG-16 model that is used in this study has undergone preliminary training using the 1.2 million pictures that are included in the ImageNet dataset. It is possible to use the weights that were learned during the training phase to a different issue in order to reduce the amount of time and money spent on calculation. The transfer learning strategy will be used on the breast cancer dataset, and the pre-trained model will serve as the basis for its application. We have a number of different options available to us when it comes to how we put the transfer learning approach to use. Some of these options include feature extraction, fine-tuning, and freezing specific layers while re-training other layers. During this experiment, we will use transfer learning to perfect the last few layers of the models by making adjustments to their parameters. In order to do this, the weights of the earliest layers of the pre-trained network will be maintained while the last few layers of the network will have their parameters adjusted.

Fine-tuning

Fine-tuning is the process of tweaking the value of the model's hyper-parameters in order to get a good model. The variables that are established before to the beginning of the training process are referred to as hyper-parameters. When it comes to achieving the best possible performance from a network, hyper-parameters play a crucial role. As a consequence of this, we need to modify these hyper-parameters in order to improve the performance of our models. The learning rate, the number of epochs, the batch size, and the choice of activation functions are some of the important hyper-parameters that will be modified for this study.

Summary

This chapter provided an overview of the methodology used in this study to analyze the cancer dataset. It

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue I January 2025

presented numerous CNN architectures, such as ResNet and VGG, as well as the emerging field of vision transformers. The chapter emphasized the significance of preprocessing processes, such as data enhancement, for ensuring the quality of datasets. The chosen methodologies seek to advance the comprehension and application of deep learning techniques in cancer classification. This chapter contributes valuable insights to the field of cancer detection and classification by laying the ground work for subsequent analysis and results.

RESULTS AND DISCUSSION

In this section, the results and discussions of the research are presented and analyzed. The chapter begins with an overview of the experimental setup, including the configuration of the models and evaluation metrics used. The results obtained from applying the selected methodologies, including various CNN architectures and vision transformers, to the cancer dataset are then thoroughly examined and discussed. The findings related to tumor detection, subtype classification, and prognosis prediction are highlighted and compared, shedding light on the performance and effectiveness of each approach. Additionally, any unexpected or interesting findings are explored, offering potential insights for future research. The chapter concludes by summarizing the main findings, addressing limitations, and suggesting further directions for exploration in the field of cancer classification and detection.

Residual networks

The capacity of ResNet to model complicated connections and capture nuanced patterns from medical images makes it a valuable tool for breast cancer categorization [2,4,27]. ResNet's deep architecture and skip connections lead to precise tumor categorization, which in turn facilitates earlier diagnosis and more tailored treatment plans for patients with breast cancer. In this study we use a ResNet with 34 convolutional layers. The composition of the architecture is described in figure 5.

Model description and advantage

We consider here a Resnet model with 34 convolutional layers (Figure 9) as it has not yet been studied for breast cancer classification. The first layer is made of a convolutional layer with 64 filters or kernels and a stride of 2 followed by a pooling operation. Then we have 16 residual blocks with two convolutions each. Finally, we have a fully connected layer with a binary output helping for our classification. In each block, we apply batch normalization after each convolution and activate with a ReLu function. The advantage of the ReLU function has been given in Section 2.2 of chapter 2. For the batch normalization, it helps to fasten the training procedure.

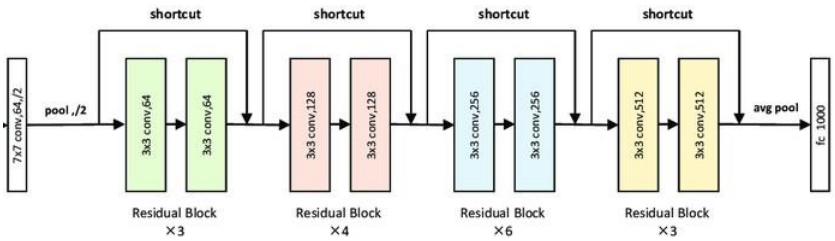

Figure 9: Structure of Resnet 34

Dataset splitting

In this case, out of a total of 277524 images, 55506 were chosen for testing and 222018 were used for training. The model was trained using around 80% of the whole data set, while the remaining 20% was utilized for testing purposes. This was done so that the test set could be kept separate (totally different from the training set). Moreover, the training phase did not make use of the data that was later used for the test set. In other words, the pictures which were used in the training phase were not employed for the testing portion of the experiment. Therefore, the effectiveness of the model was judged based on how well it prevented information from leaking out. The repartition of the positive IDC and negative IDC images is given in the

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue I January 2025

Table 1 below.

Table 1: Dataset repartition between training, testing and validation set for ResNet.

Cancer type\Sets	Training	Testing
Positive IDC	63,028	15,758
Negative IDC	158,990	39,748
Total	222,018	55,506

Model performance

When testing the model on the test set, it achieved an amazing accuracy of 90.40%. The experiment has been done in 248708.53seconds, that is quite 69 hours.

We also evaluate the model with a sample of 5555 images and got the scores in Table 2

Table 2:Performance metric on Resnet for breast cancer classification

Class\Score	Precision	Recall	F1-score
Positive Cancer	71.82%	84.46%	77.40%
Negative cancer	93.29%	87.35%	90.16%

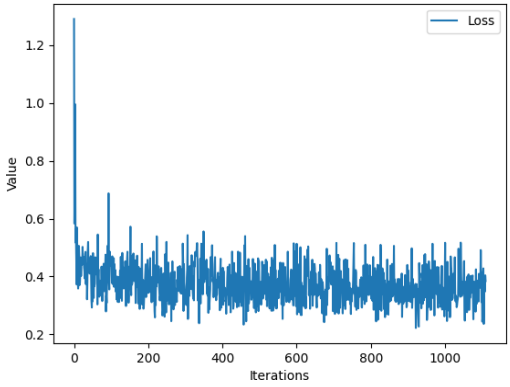

Figure 10: Evolution of ResNet loss

Discussion

The experiment with Resnet was the most accurate among the proposed models for classifying breast cancer images and achieved an accuracy score of 90.40%. The training was done without transfer learning so the weight values were learned from scratch. The experiment on the small sample image has shown a higher precision for the negative class which can be explained by the difference in number of the two classes images. The performance we obtained is satisfactory compare to other breast cancer classification-based models (99.1% for pretrained Resnet-50 [2], 98.77% for pretrained Resnet-101 [17]) because it has lower depth.

Convolutional neural network

Convolutional Neural Networks are highly effective for image classification tasks, including the classification of breast cancer images. They excel in analyzing visual data by utilizing convolutional layers to extract meaningful features from images.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue I January 2025

Model description and advantage

We utilize a convolutional network with 33 convolutions and 2 dense layers. This choice was motivated by the use of Resnet 34 in the other architecture to be able to compare the two. The depth of the architecture allows it to learn complicated patterns which makes it suitable for our classification. After every hidden layer, we use a ReLU activation function because of its ability to fasten the learning process. The dense layers are used at the end for classification and the output is activated using sigmoid. The convolutions are built with 64 filters and kernel size of 3×3.

Dataset splitting

We split the data into two subsets, namely a training and a testing set. The proportions are similar to the case of Resnet (see Table 1). We load the subsets using data loader methods from Pytorch and use batch size of 100. This method reduced considerably the running time of the model.

Model evaluation and performance

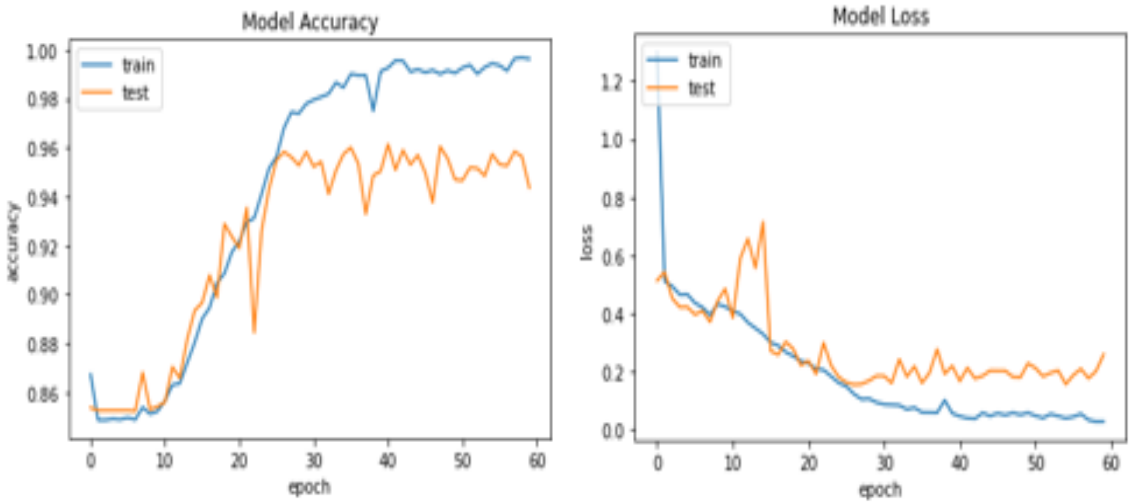

Figure 11: Accuracy and loss evolution

We used 60 epochs for training the model. The training procedure was about 6 hours of time. The model achieved an accuracy of 71.64%.

Results with VGG-16

We split the data using the ratio 70:15:15. The choice of splitting the data into 70% for training, 15% for validation, and 15% for testing is a common practice in machine learning. This split provides enough data for training, enables effective hyperparameter tuning and model selection, and allows for unbiased evaluation of the model's performance on unseen data. The training set provides a substantial amount of data for the model to learn from, the validation set aids in optimizing hyperparameters, and the testing set ensures an unbiased assessment of the model's generalization ability. By keeping the datasets separate, the integrity of the evaluation process is maintained, and the model's real-world performance can be reliably estimated.

Experimental condition

The stochastic gradient descent algorithm, with a learning rate of 0.001, was employed as the optimizer for the system. The model was tweaked to perfection using the procedures outlined in Section 3.2. For the purpose of fine-tuning, the first 15 convolutional layers of the VGG-16 model were frozen. The stochastic gradient descent optimizer was then used to train the last convolutional layer as well as the fully connected layers.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue I January 2025

Table 3: Dataset repartition between training, testing and validation set for VGG

Cancer type\Sets	Training	Testing	Validation
Positive IDC	55,150	11,818	11,818
Negative IDC	139,116	29,811	29,811
Total	194,266	41,699	41,699

Model evaluation and performance

For the training part we use a total of 50 epochs and the weight were trained using stochastic gradient descent. The evolution of the loss and accuracy during training are given in Figure 12. The model training finishes after 49 hours and achieve accuracy score of 86.81% . The precision and F1-score are given in Table 4 below.

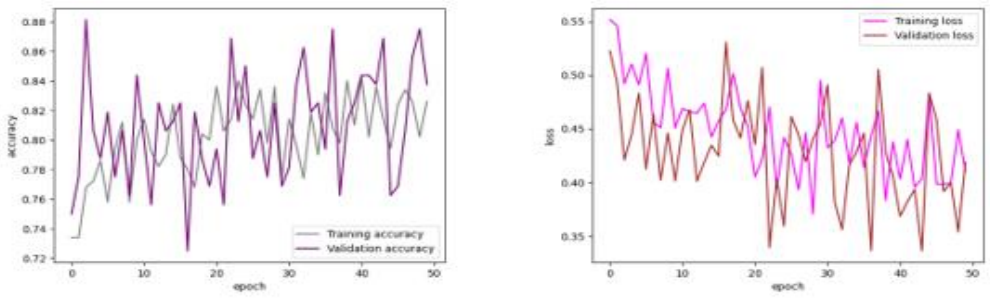

Figure 12: Training and validation loss and accuracy.

Table 4: Performance metric on VGG for breast cancer classification

Class \Score	Precision	Recall	F1-score
Positive Cancer	78%	75%	76%
Negative cancer	90%	92%	91%

Results with vision transformer

Vision transformers have several advantages for breast cancer classification, including improved accuracy compared to other models, reduced computational complexity, effective transfer learning capabilities, and flexibility in handling different types of breast cancer imaging data. These benefits make vision transformers a promising approach for enhancing breast cancer classification accuracy while maintaining computational efficiency and adaptability to diverse imaging datasets.

Model description and experimentation

The understanding of how Visions Transformers works can be seen in Section 3. The data set is split the same way like for ResNet (table 1). Table 5 displays the different parameters in our model and their value.

Table 5: Experimental condition for training

Parameters	Values
Number of epochs	10

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue I January 2025

Learning rate	0.001
Weight decay	0.0001
Batch size	10
Dimension of projection	32
Number of heads in the encoder	8
New image size	72
Patch size	6
Loss function	Sparse categorical Cross-entropy loss from Keras
Optimizer	Adam from Tensorflow-addons

The image new size is set to 72×72 due to computational and time limitation. However, there is enough patches to capture attention in each image. The batch size is 10 because in the experimentation it helps the learning to happen faster.

Model evaluation and performance

The training was achieved in 24 hours on the whole 80% of the dataset. The evaluation of the model on unseen data displayed an accuracy of 83.33%. We also use a sample of 3331 images to access the precision, recall and F1-score where the results are given in Table 6.

Table 6: Performance metric for on ViT for breast cancer classification

Class\Score	**Precision**	**Recall**	**F1-score**
Positive Cancer	62.83%	84.12%	72.20%
Negative cancer	93.25%	82.44%	87.31%

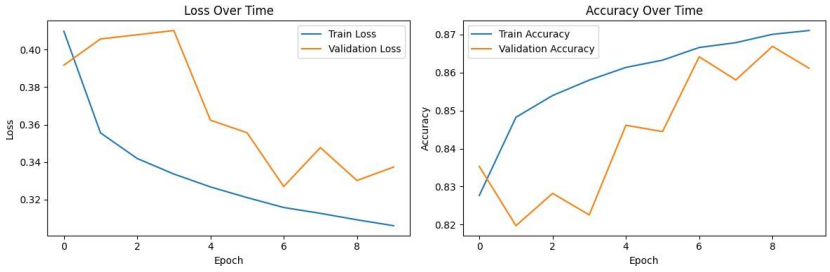

Figure 13: Accuracy and loss evolution with ViT.

Discussion

In this section, we discuss the results we have found previously.

The two graphs Figure 12 **a**nd Figure 13 show the training loss and testing loss for VGG-16 and Vision transformers respectively. The first graph (Figure 12) shows that VGG-16 is not overfitting the training data, while the second graph (Figure 13) shows that the complexity of vision transformer architecture makes it overfit the training data. This is because the Vision Transformers has more parameters to learn, and as a result, it takes longer for the model to converge. The ideal situation would be for the training loss and testing loss to decrease together, but this is not always possible. In this case, the goal is to minimize the testing loss

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue I January 2025

as much as possible and one can do that by reducing the number of parameters of the model.

Overall, the Figure 12 shows that the VGG-16 model is learning effectively, while Figure 13 shows that the ViT is overfitting the training data.

Resnet-34 appears to be learning effectively as we can see in Figure 9. This raises the model to the best model in our experiments as it exhibits the best accuracy among the proposed models. However, the power of transfer learning heights VGG-16 in the first position when analyzing F1-score, which for this case is 91% for the negative class and 76% for the positive class. This is the best model overall since we are dealing with imbalanced data. Moreover, it finishes after 49 hours.

The running time analysis of the models shows that the vision transformer model is fast in learning but it does not perform as well as Resnet and VGG. However, we will notice that the experiments have been done in different conditions. While Resnet took 69 hours to run using 50 epochs, ViT took 24 hours to complete using 10 epochs. The VGG-16 model due to transfer learning showed a rapid training process and finished in 49 hours for 50epochs.

The CNN model showed the least accuracy measure. This is because the architecture used same convolutional layers stacked together reducing the learning ability of the model.

The recall and precision recorded from the different models show the imbalance in the data. The fact that the negative class is associated with highest values of these measure is due to the higher number of negative classes. Resnet and ViT demonstrate a great ability to recognize negative class images with precision of 93%. VGG-16 performed also well with a precision of 90%.

Resnet has the highest F1-score for the positive class followed by VGG-16. This result demonstrates the power of Resnet and VGG to handle imbalanced data (Table 3).

The python codes used to generate all the results can be found here.

CONCLUSION

In this research, we provided a detailed study on the use of deep transfer learning for the classification of breast cancer. We employed a Vision Transformers, a Resnet-34, a plain CNN and a VGG-16 to classify images of invasive ductal carcinoma cancer. The result has demonstrated the power of residual networks in image recognition putting Resnet-34 at the top with an accuracy of 90%. The network structure with skip connection allowed an effective learning of image features. However, due to the imbalanced data, we shall rely on the F1-score measure for which VGG outperformed. We noticed that the use of transfer learning is required for fast training and better generalization of models. Vision transformer demonstrated its power in classifying images. However, the model experienced overfitting due to the multitude of parameters it takes in account. The VGG-16 showed a good learning trend due to the fact that the model has been pretrained and therefore recognize some feature already which makes it to update quickly the weights of the network and minimize the error in predictions.

Future work

We did not perform all the experiments in uniform experimental conditions by the fact of unavailability of strong computational resources and time limitation. So, we exhibit some plan for future works:

Investigate the use of a new loss function to improve the accuracy of the models.

Perform detection and multi-class classification of breast cancer abnormalities.

REFERENCES

1. Abdolahi, M., Salehi, M., Shokatian, I., & Reiazi, R. (2020). Artificial intelligence in automatic

classification of invasive ductal carcinoma breast cancer in digital pathology images. Medical Journal of the Islamic Republic of Iran, 34, 140.
2. Al-Haija, Q., & Adebanjo, A. (2020). Breast cancer diagnosis in histopathological images using resnet-50 convolutional neural network. In IEEE International IOT, Electronics and Mechatronics Conference (pp. 1-7).
3. AlGhodhaifi, H., AlGhodhaifi, A., & AlGhodhaifi, M. (2019). Predicting invasive ductal carcinoma in breast histology images using convolutional neural network. In IEEE National Aerospace and Electronics Conference (pp. 374-378).
4. Baccouche, A., Garcia-Zapirain, B., & Elmaghraby, A. (2022). An integrated framework for breast mass classification and diagnosis using stacked ensemble of residual neural networks. Scientific Reports, 12(1), 1-17.
5. Baevski, A., & Auli, M. (2018). Adaptive input representations for neural language modeling. In International Conference on Learning Representations.
6. Banjara, B. Deep residual learning for image recognition (resnet explained). Retrieved from https://bit.ly/3JNTWY7
7. Boesch, G. Vgg very deep convolutional networks (vggnet) – what you need to know. Retrieved from https://viso.ai/deep-learning/vgg-very-deep-convolutional-networks/
8. Bolhasani, H., Amjadi, E., Tabatabaeian, M., & Jassbi, S. J. (2020). A histopathological image dataset for grading breast invasive ductal carcinomas. Informatics in Medicine Unlocked, 19.
9. Budak, Ü., Cömert, Z., Rashid, Z., Şengür, A., & Çıbuk, M. (2019). Computer-aided diagnosis system combining FCN and bi-LSTM model for efficient breast cancer detection from histopathological images. Applied Soft Computing, 85.
10. Chhikara, B., & Parang, K. (2023). Global cancer statistics 2022: The trends projection analysis. Chemical Biology Letters, 10, 451-451.
11. Code, P. W. Vgg. Retrieved from https://paperswithcode.com/method/vgg
12. Das, A., Mohanty, M. N., Mallick, P. K., Tiwari, P., Muhammad, K., & Zhu, H. (2021). Breast cancer detection using an ensemble deep learning method. Biomedical Signal Processing and Control, 70.
13. Datagen. Understanding vgg16: Concepts, architecture, and performance. Retrieved from https://datagen.tech/guides/computer-vision/vgg16/
14. Deepchecks. Vggnets. Retrieved from https://deepchecks.com/glossary/vggnet/
15. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Gelly, S. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.
16. Faghihroohi, S. Vision transformers. Retrieved from https://wiki.tum.de/display/dlma/Vision+Transformers
17. Gandomkar, Z., Brennan, P., & Mello-Thoms, C. (2018). MuDeRN: Multi-category classification of breast histopathological image using deep residual networks. Artificial Intelligence in Medicine, 88, 14-24.
18. GeeksforGeeks. Vgg-16 & cnn model. Retrieved from https://www.geeksforgeeks.org/vgg-16-cnn-model/
19. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In IEEE Conference on Computer Vision and Pattern Recognition (pp. 770-778).
20. Indolia, S., Goswami, K., Mishra, P., & Asopa, P. (2018). Conceptual understanding of convolutional neural network-a deep learning approach. Procedia Computer Science, 132, 679-688.
21. Krizhevsky, A., Sutskever, I., & Hinton, G. (2017). Imagenet classification with deep convolutional neural networks. Communications of the ACM, 60, 84-90.
22. Li, Q., Cai, W., Wang, X., Zhou, Y., Feng, D., & Chen, M. (2014). Medical image classification with convolutional neural network. In 13th International Conference on Control Automation Robotics & Vision (pp. 844-848).
23. Nasser, M., & Yusof, K. (2023). Deep learning based methods for breast cancer diagnosis: A systematic review and future direction. Diagnostics, 13(1), 161.
24. Nusrat, A. B., Aimon, R., & Mahdy, M. (2021). Automated detection and grading of invasive ductal carcinoma breast cancer using ensemble of deep learning models. Computers in Biology and Medicine, 139, 104931.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue I January 2025

25. Phil, K. (2017). Matlab deep learning with machine learning, neural networks and artificial intelligence. Apress.
26. Pratiwi, K., Fu'adah, Y., Ibrahim, N., Rizal, S., & Saidah, S. (2021). Invasive ductal carcinoma (IDC) classification based on breast histopathology images using convolutional neural network. In 1st International Conference on Electronics, Biomedical Engineering, and Health Informatics (pp. 477-486).
27. Pullaiah, N., Venkatasekhar, D., & Venkatramana, P. (2022). Breast cancer classification using customized resnet based convolution neural networks. In 2022 International Conference on Knowledge Engineering and Communication Systems (pp. 1-5).
28. Refaeilzadeh, P., Tang, L., & Liu, H. (2009). Cross-validation. Encyclopedia of Database Systems, 532-538.
29. Singh, A., Arjunaditya, & Tripathy, B. K. (2023). Detection of Cancer Using Deep Learning Techniques. Springer Nature Singapore.
30. Sonka, M., Hlavac, V., & Boyle, R. (2014). Image processing, analysis, and machine vision. Cengage Learning.
31. Sung, H., Ferlay, J., Siegel, R., Laversanne, M., Soerjomataram, I., Jemal, A., & Bray, F. (2021). Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. CA: A Cancer Journal for Clinicians, 71(3), 209-249.
32. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., ... & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30.
33. Wang, Q., Li, B., Xiao, T., Zhu, J., Li, C., Wong, D., & Chao, L. (2019). Learning deep transformer models for machine translation. arXiv preprint arXiv:1906.01787.
34. Zhang, Z., Wang, S., Li, Z., Gao, F., & Wang, H. (2023). A multi-dimensional covert transaction recognition scheme for blockchain. Mathematics, 11(4), 1015