Classification of Histopathological Breast cancer images using Deep Learning

Rishab Kumar, 1701EE39

Under the guidance of: **Dr. M H Kolekar**

1. Abstract

Breast cancer (BC) infection, which is peculiar to women, brings about the high rate of deaths among women in every part of the world. Therefore, the early and precise diagnosis of breast cancer plays a pivotal role to improve the prognosis of patients with this disease. Several studies have developed automated techniques using different medical imaging modalities to predict breast cancer development. Deep learning can extract and organize the discriminative information from the data, not requiring the design of feature extractors by a domain expert. Convolutional Neural Networks (CNNs) are a particular type of deep, feedforward network that have gained attention from research community and industry, achieving empirical successes in tasks such as speech recognition, signal processing, object recognition, natural language processing and transfer learning. I propose a model which is able to extract features from breast histopathological images using 5 different independent models and then combine all the features to have a wide variety of features and hence the classification is more efficient.

2. Introduction

There are many types of datasets capturing the data of breast cancer. These are histopathological images, mammogram images, extracted feature based data, ultrasounds, MRI images, etc. In this work, I will be working on histopathological breast cancer images from different standard datasets. Histopathology refers to the microscopic examination of tissue in order to study the manifestations of disease. Recently, deep learning-based approaches have been shown to outperform conventional machine learning methods, allowing automation of end-to-end processing and hence it is obvious to use deep learning models to achieve promising results. Different efficient pre build deep learning models are available which work differently in different circumstances. The feature extraction for different varies according to the type of dataset and hence I have proposed a model in which feature will be extracted from 5 different models and will be combined afterwards. Since features are extracted from different models independently, hence the chances of missing any major feature is very less. The method is bound to have better accuracy than using a single model for the classification.

3. Related Work

Significant work have been made both in machine learning and deep learning to develop methods for classification of histopathological breast cancer images. Before the deep learning revolution, machine learning approaches, including support vector machine (SVM), principle component analysis (PCA), and random forest (RF) methods were used to examine the data. Although the results from machine learning models are not impressive and there is a need to develop a deep learning model as the images contain complex features which are difficult to extract using machine learning models. Several research papers have worked on the Breakhis dataset **[10]** for all the four magnification factors that is 40x, 100x, 200x and 400x. Certain state-of-the-art algorithms claim their accuracies on the Breakhis dataset **[10]** for binary classification as referenced from **[3]**. It tested many models with the Breakhis dataset and highest accuracies of all the models were 85.36%, 91% and 92.19% The paper **[1]** has achieved highest accuracy of 90%, 90%, 91%, 90% for 40x, 100x, 200x and 400x respectively. Many deep learning models **[2], [4], [5]** have achieved accuracies from 90% - 94% on the same dataset.

4. Datasets

There are many datasets available in the field of breast cancer histopathological images. In this work, I have used 3 standard breast cancer histopathology image datasets which are available publicly for research work. Any breast tissue can be classified into 2 classes i.e tumour and non-tumour. The tumour images can be classified into cancerous tumour (malignant) and non-cancerous tumour (benign).The benign images can be further classified into in-situ and invasive.

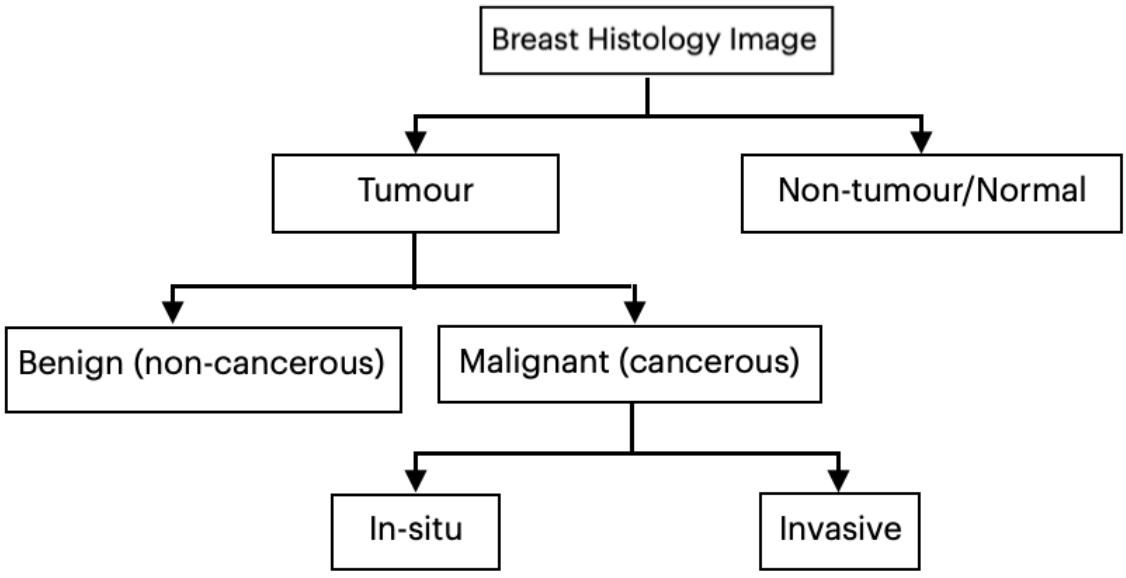


Figure 1: Flow chart showing classification of breast image to different categories

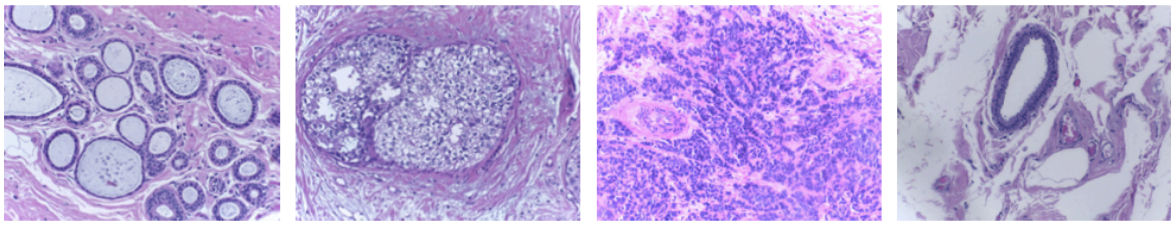
4.1 ICIAR2018: This dataset **[12]** (publicly available at ***https://iciar2018-challenge.grand-challenge.org/Dataset/***) is an extended version of the Bioimaging 2015 breast histology classification challenge dataset. All of the images were digitised under the same acquisition conditions; the magnification is 200× and the pixel dimensions are 0.420 µm × 0.420 µm. As shown in Figure 2, each image is categorised into one of four classes: (1) benign, (2) in situ, (3) invasive carcinoma, and (4) normal; for each case, the assigned class corresponds to a predominant cancer type in the respective image. The goal of this challenge was to provide an automatic classification of each input image

The dataset contains a total of 400 microscopy images, distributed as follows:

- *Normal: 100*
- *Benign: 100*
- *in situ: 100*
- *Invasive: 100*

Microscopy images are on .tiff format and have the following specifications:

- *Color model: R(ed)G(reen)B(lue)*
- *Size: 2048 x 1536 pixels*
- *Pixel scale: 0.42 µm x 0.42 µm*
- *Memory space: 10-20 MB (approx.)*
- *Type of label: image-wise*



(a) Benign (b) InSitu (c) Invasive (d) Normal

Figure 2: histopathology images in the dataset ICIAR2018

4.2 Breakhis: This dataset ***[10]***(publicly available at ***https://web.inf.ufpr.br/vri/databases/breast-cancer-histopathological-database-breakhis/***) is composed of 9,109 microscopic images of breast tumour tissue collected from 82 patients using different magnifying factors (40X, 100X, 200X, and 400X). To date, it contains 2,480 benign and 5,429 malignant samples (700X460 pixels, 3-channel RGB, 8-bit depth in each channel, PNG format). the size of each image is 700 × 460 pixels. Each class has four subclasses, with the four types of benign tumors being adenosis, fibroadenoma, tubular adenoma, and phyllodes tumor. The four subclasses of cancer are ductal carcinoma, lobular carcinoma, mucinous carcinoma, and papillary carcinoma. This database has been built in

collaboration with the P&D Laboratory – Pathological Anatomy and Cytopathology, Parana, Brazil.

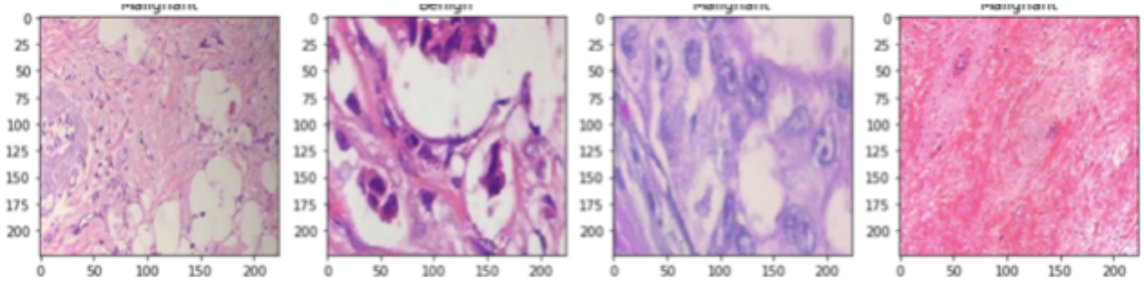


Figure 3: Two types of benign and malignant tumour images from the Breakhis dataset. The magnification factor of these images is 40×.

Classes	Subtypes	Magnifcation Factors				Total	# of Patients
		40×	100×	200×	400×		
Benign	Adenosis (A)	114	113	111	106	444	4
	Fibroadenoma (F)	253	260	264	237	1014	10
	Tubular Adenona (TA)	109	121	108	115	453	3
	Phyllodes Tumor (PT)	149	150	140	130	569	7
Malignant	Ductal Carcinoma (DC)	864	903	896	788	3451	38
	Lobular Carcinoma (LC)	156	170	163	137	626	5
	Mucinous Carcinoma (MC)	205	222	196	169	792	9
	Papillary Carcinoma (PC)	145	142	135	138	560	6
Total		1995	2081	2013	1820	7909	82

Figure 4: Information about distribution of BreakHis dataset images

4.3 Kaggle Breast Histology Dataset: The original dataset **[11]** consisted of 162 whole mount slide images of Breast Cancer (BC) specimens scanned at 40x. This dataset consists a total of 277,524 images divided into 198,738 benign and 78,786 malignant images. The original dataset is available publicly at ***https://www.kaggle.com/paultimothymooney/breast-histopathology-images***

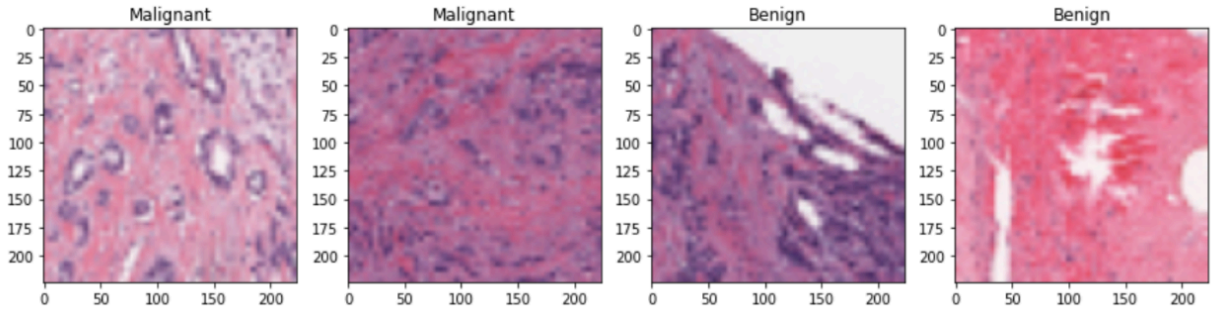


Figure 5: Two types of benign and malignant tumour images from the Kaggle Breast Histology dataset. The magnification factor of these images is 40×.

5. Methodology

CNN-based methods have various strategies to increase the performance of image classification on small datasets. There are certain other processes used such as batch normalisation and data augmentation. Therefore a deep neural network is used which consists of convolutional layers which has the power to classify images which are difficult for human eyes as deep convolutional layers extract complex hidden features present in histopathological images. Classifying images with one architecture can give good results but a single architecture is not able to capture every feature on its own, every single architecture misses some features and hence it is not easy to achieve a great efficiency by training on a single architecture.
I have proposed a model where 5 most used classification modules are connected in parallel and the output of all the five models are concatenated in order to unionise all the features extracted be different models. Hence combining 5 independent modules and concatenating them reduces the chance of missing any relevant features.

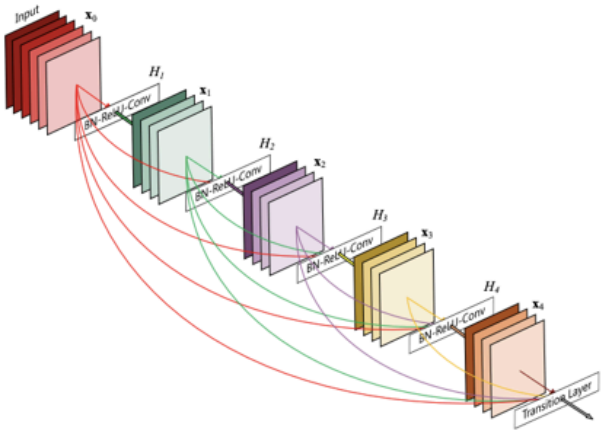
5.1. Implementation Details

The model consists of 5 most used classification modules named DenseNet201, VGG19, ResNet152, Xception and Inception_v3. These 5 prebuilt architectures are used in classification techniques in many areas. These are the best performing deep learning architectures in the areas of classification tasks.

5.1.1 DenseNet

The Dense Convolutional Network (DenseNet), which connects each layer to every other layer in a feed-forward fashion. Whereas traditional convolutional networks with L layers have L connections - one between each layer and its subsequent layer - our network has L(L+1)/2 direct connections. For each layer, the feature-maps of all preceding layers are used as inputs, and its own feature-maps are used as inputs into all subsequent layers. DenseNets have several compelling advantages: they alleviate the vanishing-gradient problem, strengthen feature propagation, encourage feature reuse, and substantially reduce the number of parameters. We evaluate our proposed architecture on four highly competitive object recognition benchmark tasks (CIFAR-10, CIFAR-100, SVHN, and ImageNet).

I have used DenseNet201 because it has shown magnificent results in certain image classification problems. In a DenseNet201 the error signal can be easily propagated to



DenseNet Structure

$$a^{[l]} = g\left([a^{[0]}, a^{[1]}, a^{[2]}, \ldots\ldots\ldots\ldots, a^{[l-1]}\right)$$

Figure 6: Structure of DenseNet

earlier layers more directly. This is a kind of implicit deep supervision as earlier layers can get direct supervision from the final classification layer.

5.1.2. VGG

VGG is a popular neural network architecture proposed by Karen Simonyan & Andrew Zisserman from the University of Oxford. It is also based on CNNs, and was applied to

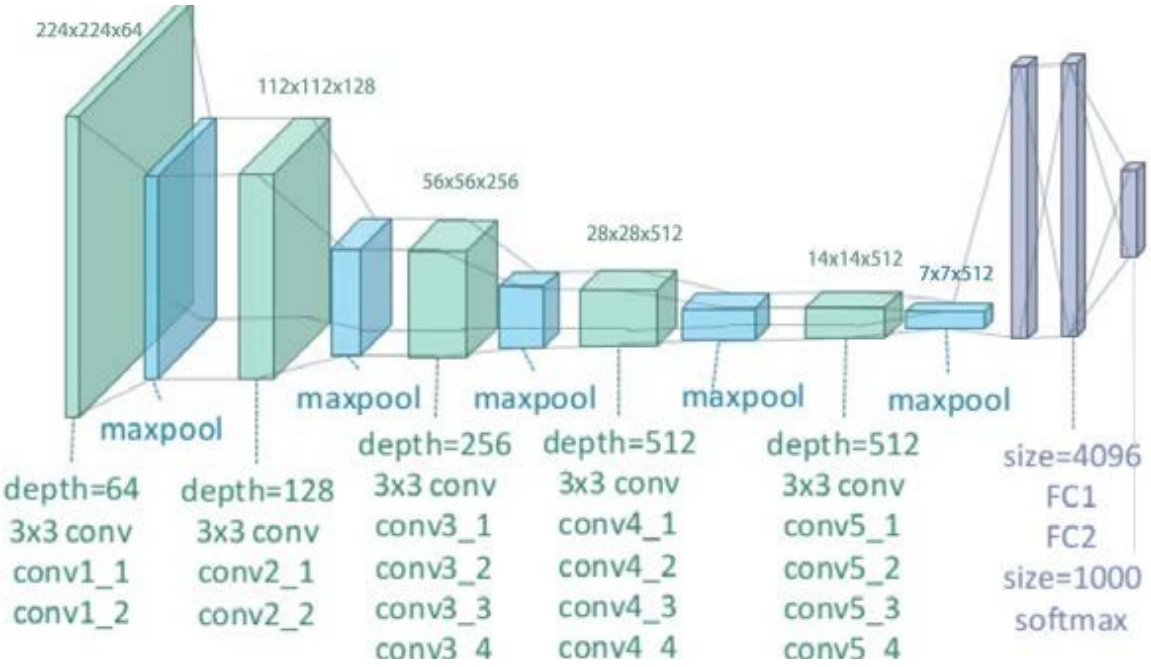


Figure 7: Structure of VGG

the ImageNet Challenge in 2014. The network achieved 92.7% top-5 test accuracy on the ImageNet dataset.

Major improvements of VGG, when compared to AlexNet, include using large kernel-sized filters (sizes 11 and 5 in the first and second convolutional layers, respectively) with multiple (3×3) kernel-sized filters, one after another.

5.1.3. ResNet

ResNet, short for Residual Network is a specific type of neural network that was introduced in 2015 by Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun in their paper “Deep Residual Learning for Image Recognition”.

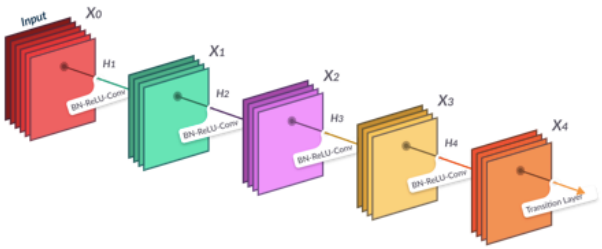


Figure 8: Structure of ResNet

Mostly in order to solve a complex problem, we stack some additional layers in the Deep Neural Networks which results in improved accuracy and performance. The intuition behind adding more layers is that these layers progressively learn more complex features.

5.1.4. Xception

Xception is a convolutional neural network that is 71 layers deep. You can load a pretrained version of the network trained on more than a million images from the ImageNet database [1]. The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals. As a result, the network has learned rich feature representations for a wide range of images. The network has an image input size of 299-by-299.

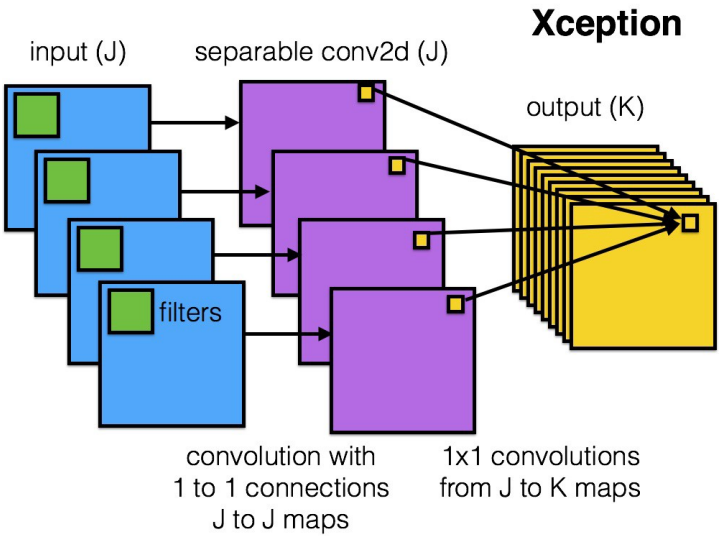


Figure 9: Structure of Xception neural network

5.1.5. Inception_v3

Inception v3[1] is a convolutional neural network for assisting in image analysis and object detection, and got its start as a module for Googlenet. It is the third edition of Google's Inception Convolutional Neural Network, originally introduced during the ImageNet Recognition Challenge. Just as ImageNet can be thought of as a database of classified visual objects, Inception helps classification of objects[2] in the world of computer vision.

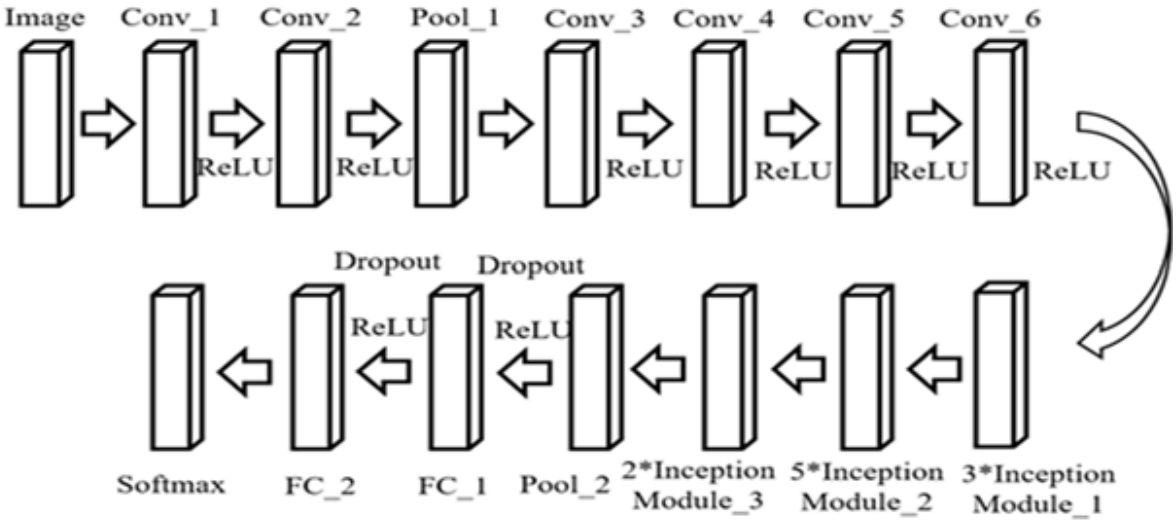


Figure 10: Structure of Inception_v3 neural network

5.1.2 Model Details

The size of input image is taken as (224 x 224 x 3) which comprises of RGB image of hight and width of 224 pixels. After the DenseNet201, there is a global Average pooling layer which is followed by a dropout layer whose dropout factor is set to be 0.5, It will randomly select 50% of neurons and ignore them during training. This is done to reduce the complexity of the model and reduce the training time. The dropout layer is followed by a batch normalisation layer which is later followed by a dense layer which consists of an activation function. Softmax function is chosen as the activation function in the dense layer. The last layer is this dense layer which has two outputs either of which will be depending on the prediction. The loss function used in the model is binary cross-entropy and the optimiser use is Adam optimiser. It contains 2 modules, one for feature extraction and the other for classification. The features from the 5 different models are concatenated and then transferred to classification module. The classification module contains batch normalisation layers, dense layers and some dropout layers for classification. The DenseNet201 architecture contains 18321984 parameters, VGG19 architecture contains 20024384 parameters, ResNet152 architecture contains 58370944 parameters, Xception architecture contains 20861480 parameters and inception_v3 architecture contains 21802784 parameters. The complete model contains a total of 148,509,290 trainable parameters.

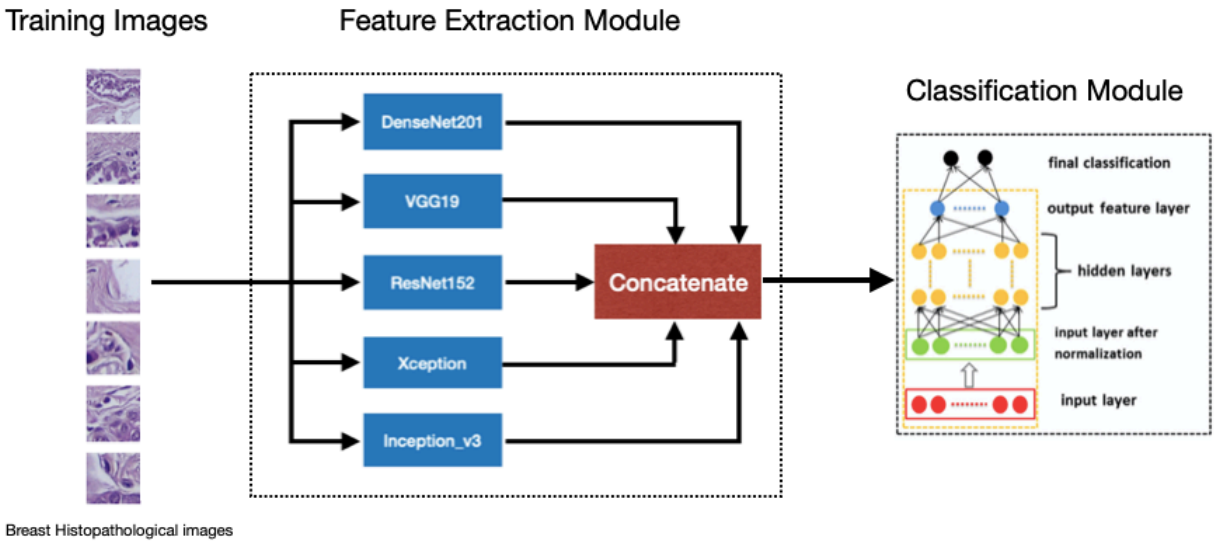


Figure 11: Flowchart of the proposed methodology

The input images are fed to all the 5 models independently which are then concatenated to combine all the features and then the features are transferred to the classification module which is responsible for classifying. There are certain dropout layers too which are required to remove extra features which increases the complexity of the model. Below is the diagram of the summary of the model. It describes different layers in the model, and the output shape of all the layers.

Model: "model"

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 224, 224, 3)	0	
densenet201 (Functional)	(None, 7, 7, 1920)	18321984	input_1[0][0]
vgg19 (Functional)	(None, 7, 7, 512)	20024384	input_1[0][0]
resnet152 (Functional)	(None, 7, 7, 2048)	58370944	input_1[0][0]
xception (Functional)	(None, 7, 7, 2048)	20861480	input_1[0][0]
global_average_pooling2d (Globa	(None, 1920)	0	densenet201[0][0]
global_average_pooling2d_3 (Glo	(None, 512)	0	vgg19[0][0]
global_average_pooling2d_4 (Glo	(None, 2048)	0	resnet152[0][0]
global_average_pooling2d_1 (Glo	(None, 2048)	0	xception[0][0]
inception_v3 (Functional)	(None, 2048)	21802784	input_1[0][0]
dropout (Dropout)	(None, 1920)	0	global_average_pooling2d[0][0]
dropout_3 (Dropout)	(None, 512)	0	global_average_pooling2d_3[0][0]
dropout_4 (Dropout)	(None, 2048)	0	global_average_pooling2d_4[0][0]
dropout_1 (Dropout)	(None, 2048)	0	global_average_pooling2d_1[0][0]
dropout_2 (Dropout)	(None, 2048)	0	inception_v3[0][0]
batch_normalization (BatchNorma	(None, 1920)	7680	dropout[0][0]
batch_normalization_101 (BatchN	(None, 512)	2048	dropout_3[0][0]
batch_normalization_102 (BatchN	(None, 2048)	8192	dropout_4[0][0]
batch_normalization_5 (BatchNor	(None, 2048)	8192	dropout_1[0][0]
batch_normalization_100 (BatchN	(None, 2048)	8192	dropout_2[0][0]
concatenate_2 (Concatenate)	(None, 8576)	0	batch_normalization[0][0] batch_normalization_101[0][0] batch_normalization_102[0][0] batch_normalization_5[0][0] batch_normalization_100[0][0]
dense (Dense)	(None, 1000)	8577000	concatenate_2[0][0]
dropout_5 (Dropout)	(None, 1000)	0	dense[0][0]
dense_1 (Dense)	(None, 1000)	1001000	dropout_5[0][0]
dropout_6 (Dropout)	(None, 1000)	0	dense_1[0][0]
dense_2 (Dense)	(None, 2)	2002	dropout_6[0][0]

Total params: 148,995,882
Trainable params: 148,509,290
Non-trainable params: 486,592

Figure 12: Architecture of the proposed model

5.2 Training

The model is trained and tested on all the three mentioned datasets. 80% of the data is used for training and 20% data is used for testing the model. A batch size of 16 is used and epoch of 100 is used to train the dataset. A learning rate of 0.0001 is used.

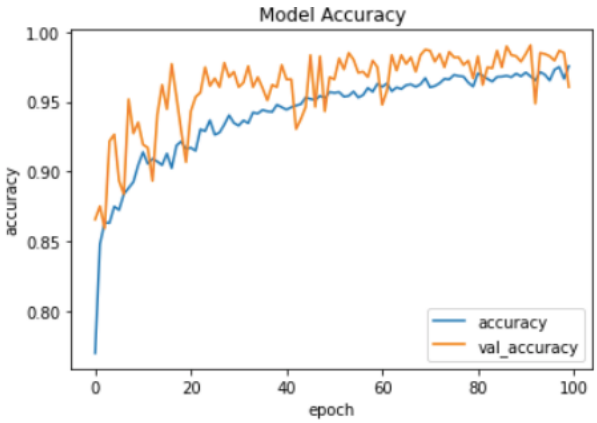


Figure 13: Accuracy and validation accuracy for 100 epochs

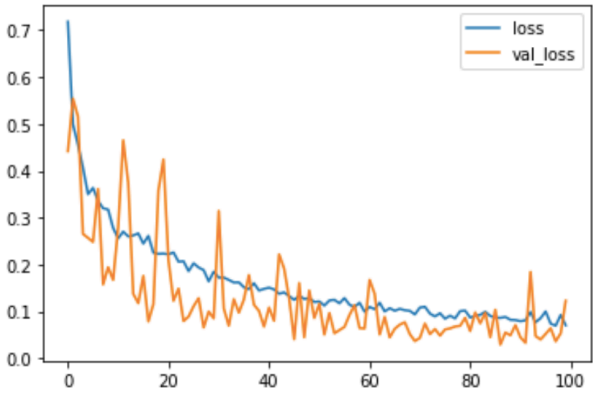


Figure 14: Loss and validation loss for 100 epochs

The Breakhis dataset **[10]** took 32 hours to train, the Kaggle BHI **[11]** took 10 hours to train and ICIAR2018 dataset **[12]** took 2 hours to train. Since the Breakhis dataset **[10]** is distributed into four different divisions as the magnification factors of 40x, 100x, 200x and 400x. The testing is done separately for all 4 magnifications to get the individual results. The Breakhis dataset contains a total of 7909 images, the ICIAR2018 contains a total of 400 images before data augmentation and the Kaggle BHI contains 2401 images

	Dataset Name	TRAIN		TEST	
		Benign	Malignant	Benign	Malignant
Breakhis	40x	501	1097	126	275
	100x	517	1151	129	288
	200x	500	1113	125	279
	400x	472	987	118	247

Figure 15: Information about distribution of different dataset into training and testing images

5.3 Testing

20% of the images are used for testing purposes. Figure 11 contains the information about the number of images used in training and testing for all three datasets. The splitting of dataset into training and testing samples is done manually. The ICIAR2018 dataset contains only 400 images, hence the model could not work properly for this dataset due to the limitation of images.

6. Results

The model is tested for the testing dataset and certain performance quantities are calculated for each dataset. A performance matrix is also calculated to find out the sensitivity (recall), specificity, precision and the F-score. For any classification algorithm, the efficiency is calculated using these numbers. Hence all the quantities are calculated for all the 3 datasets, including 4 different magnifications of BreakHis dataset. All the calculations are made considering the malignant images to be positive. The performance metrics contains 4 numbers which are:

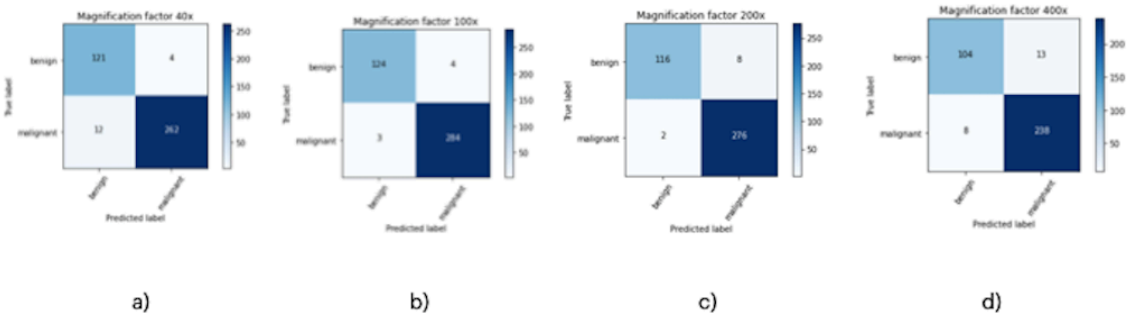


Figure 16: Confusion Matrix for different magnification factors for Breakhis dataset
a) 40x b) 100x c) 200x d) 400x

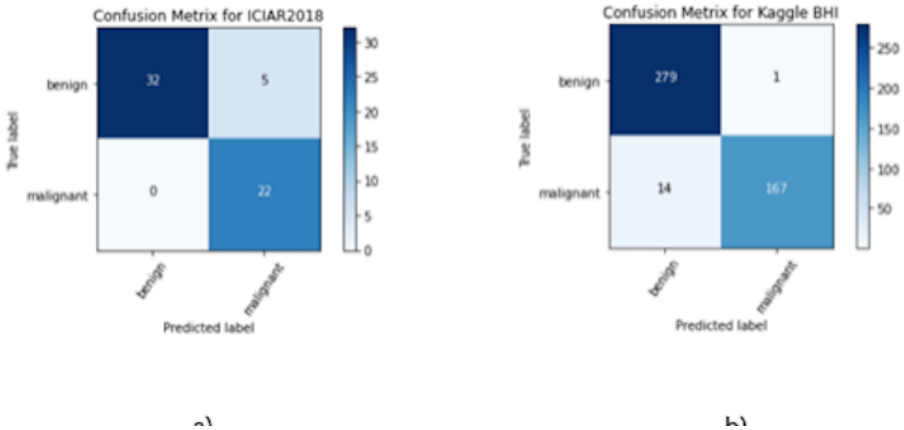


Figure 17: Confusion Matrix for a) ICIAR2018 dataset b) Kaggle BHI dataset

1. ***True Positive(TP)***: The number of observations the model predicted were positive that were actually positive.
2. ***True Negative(TN)***: The number of observations the model predicted were negative that were actually negative.
3. ***False Positive(FP)***: The number of observations the model predicted were positive that were actually negative
4. ***False Negative(FN)***: The number of observations the model predicted were negative that were actually positive.

The formula for different performance quantities are as follows:

$$Accuracy = \frac{(TP + TN)}{(TP + FP + FN + TN)} \qquad Recall = \frac{TP}{(TP + FN)}$$

$$Precision = \frac{TP}{(TP + FP)} \qquad Specificity = \frac{TN}{(TN + FP)}$$

$$Sensitivity = \frac{TP}{(TP + FN)} \qquad F1 - Score = 2 * \frac{Recall * Precision}{Recall + Precision}$$

Dataset Name		Accuracy	Sensitivity/ Recall	Specificity	Precision	F1-score
Breakhis	40x	95.98%	98.49%	90.97%	95.62%	0.97
	100x	98.31%	98.61%	97.63%	98.95%	0.98
	200x	97.51%	97.18%	98.30%	99.28%	0.98
	400x	94.21%	94.82%	92.85%	96.74%	0.95
ICIAR 2018		91.52%	81.48%	100%	100%	0.90
KAGGLE BHI		96.74%	99.40%	95.22%	92.26%	0.96

Figure 18: Information about all the performance values (accuracy, sensitivity (recall), specificity, ` precision and F1-score for different datasets

7. Conclusions

I proposed the parallel model for classification of histological images by training different-scale image patches to learn the overall structures and texture features of cells. The model performance and strength were evaluated on three publicly available benchmark datasets, and for a variety of experimental strategies, such as multiple magnification factors, and either with or without data augmentation.

References	40x	100x	200x	400x
Kahya et al. (2017)	94.97	93.62	94.54	94.42
Wei et al. (2017)	**97.89**	97.64	97.50	**97.97**
Pratiher et al. (2018)	96.8	98.1	98.2	97.5
Bardou et al. (2018)	96.82	96.96	96.36	95.97
Bardou et al. (2018)	92.71	93.75	92.72	92.12
Present Work	95.98	**98.31**	**97.51**	94.21

The model is able to get good results and it outperformed some previous classification models in certain performance values such as accuracy, sensitivity, and specificity. As compared to different models in the previous table, it can be seen then the present work is performing quite good as compared to other algorithms. It is surely performing better on the 100x and 200x as it achieved 98.31% and 97.51% on 100x and 200x respectively. For the ICIAR2018 dataset, the model didn't achieved good accuracy because of less number of images to train. The ICIAR2018 dataset **[12]** contains only 400 images and the model did not fit well for such a small dataset. Although after increasing the number of images by using data augmentation. The training data was increased 4 times and hence the accuracy increased by approximately 1%. The proposed model also achieved good accuracy of 94.14% on Kaggle Breast Histology Image dataset **[11]**. The Kaggle BHI dataset is very big as it contains 2,77,524 images hence training on the full dataset is not possible on a CPU, therefore a subset of the dataset was used for training and testing purposes. The proposed model achieved good sensitivity and specificity for different cases, which is useful for pathologists and researchers working in the field of cancer diagnosis using histological images.

8. References

1. *Abdullah-Al Nahid; Mohamad Ali Mehrabi; Yinan Kong*, Histopathological Breast Cancer Image Classification by Deep Neural Network Techniques Guided by Local Clustering, *BioMed Research International* ***2018.*** *https://doi.org/10.1155/2018/2362108*

2. *Spanhol; Oliveira; Petitjean; Heutte:* Breast cancer histopathological image classification using convolutional neural networks. *In Proceedings of the 2016 International Joint*

*Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, 24–29 July **2016**; pp. 2560–2567.*

3. *Ghulam Murtaza1; Liyana Shuib1; Ainuddin Wahid Abdul Wahab1; Ghulam Mujtaba2; Ghulam Mujtaba3; Ghulam Raza4 and Nor Aniza Azmi5:* Breast cancer classification using digital biopsy histopathology images through transfer learning. *Journal of Physics: Conference Series, Volume 1339, International Conference Computer Science and Engineering (IC2SE) 26–27 April **2019**, Padang, Indonesia, Ghulam Murtaza et al 2019 J. Phys.: Conf. Ser. 1339 012035*

4. *Murtaza, G., Shuib, L., Mujtaba, G. et al.* Breast Cancer Multi-classification through Deep Neural Network and Hierarchical Classification Approach. *Multimed Tools Appl 79, 15481–15511 (**2020**). https://doi.org/10.1007/s11042-019-7525-4*

5. *Murtaza, G., Shuib, L., Abdul Wahab, A.W. et al.* Deep learning-based breast cancer classification through medical imaging modalities: state of the art and research challenges. *Artif Intell Rev 53, 1655–1720 (**2020**). https://doi.org/10.1007/s10462-019-09716-5*

6. *Assiri, A.S.; Nazir, S.; Velastin, S.A.* Breast Tumor Classification Using an Ensemble Machine Learning *Method. J. Imaging **2020**, 6, 39. https://doi.org/10.3390/jimaging6060039*

7. *Chaurasia V, Pal S, Tiwari B.* Prediction of benign and malignant breast cancer using data mining techniques. Journal of Algorithms & Computational Technology. *June **2018**:119-126. doi:10.1177/1748301818756225*

8. *Sheikh, T.S.; Lee, Y.; Cho, M.* Histopathological Classification of Breast Cancer Images Using a Multi-Scale Input and Multi-Feature Network. *Cancers **2020**, 12, 2031. https://doi.org/10.3390/cancers12082031*

9. *Spanhol, F., Oliveira, L. S., Petitjean, C., Heutte, L., A* Dataset for Breast Cancer Histopathological Image Classification, IEEE Transactions on Biomedical Engineering *(TBME), 63(7):1455-1462, **2016**.*

10. *Paul Mooney; kaggle.com:* Breast Histopathology Images, 198,738 IDC(-) image patches; 78,786 IDC(+) image patches, ***2018.***

11. *Guilherme Aresta, Daniel Riccio, Yaqi Wang, Lingling Sun, Kaiqiang Ma, Jiannan Fang, Ismael Kone, Lahsen Boulmane, Aurélio Campilho, Catarina Eloy, António Polónia, Paulo Aguiar*, BACH: Grand challenge on breast cancer histology images, Medical Image Analysis, ***2019**, ISSN 1361-8415, https://doi.org/10.1016/j.media.2019.05.010*.

12. Kahya MA, Al-Hayani W, Algamal ZY. Classification of breast cancer histopathology images based on adaptive sparse support vector machine. Journal of Applied Mathematics and Bioinformatics. 2017; 7 (1):49.

13. Wei B, Han Z, He X, Yin Y. Deep learning model based breast cancer histopathological image classifi- cation. In: Cloud Computing and Big Data Analysis (ICCCBDA), 2017 IEEE 2nd International Confer- ence on. IEEE; 2017. p. 348–353.

14.Pratiher S, Chattoraj S. Manifold Learning & Stacked Sparse Autoencoder for Robust Breast Cancer Classification from Histopathological Images. arXiv preprint arXiv:180606876. 2018;.

15.Bardou D, Zhang K, Ahmad SM. Classification of Breast Cancer Based on Histology Images Using Con- volutional Neural Networks. IEEE Access;PP(99):1–1.