DEEP LEARNING FOR SIGNAL AND IMAGE PROCESSING

BREAST CANCER HISTOLOGY DATASET ANALYSIS USING DEEP LEARNING

DONE BY

ROHITH G

ABSTRACT

Breast cancer is one of the main causes of cancer death worldwide. Early diagnostics significantly increases the chances of correct treatment and survival, but this process is tedious and often leads to a disagreement between pathologists. Computer-aided diagnosis systems showed potential for improving diagnostic accuracy.

In this work, we develop the computational approach based on deep convolution neural networks for breast cancer histology image classification. Hematologist and eosin-stained breast histology microscopy image datasets are provided as a part of the ICIAR 2018 Grand Challenge on Breast Cancer Histology Images. Our approach utilizes several deep neural network architectures and gradient boosted trees classifier.

Keywords: Medical imaging, Computer-aided diagnosis (CAD), Computer vision, Image recognition, Deep learning.

INTRODUCTION

Breast cancer is the most common cancer diagnosed among US women (excluding skin cancers). Breast tissue biopsies allow the pathologists to histologically assess the microscopic structure and elements of the tissue. Histopathology aims to distinguish between normal tissue, non-malignant (benign), and malignant lesions (carcinomas) and to perform a prognostic evaluation. A combination of hematoxylin and eosin (H&E) is the principal stain of tissue specimens for routine histopathological diagnostics.

There are multiple types of breast carcinomas that embody characteristic tissue morphology. Breast carcinomas arise from the mammary epithelium and cause a pre-malignant epithelial proliferation within the ducts, called ductal carcinoma in situ. Invasive carcinoma is characterized by the cancer cells gaining the capacity to break through the basal membrane of the duct walls and infiltrate into surrounding tissues. The morphology of tissue, cells, and subcellular compartments is regulated by complex biological mechanisms related to cell differentiation, development, and cancer.

Traditionally, morphological assessment and tumor grading were visually performed by the pathologist, however, this process is tedious and subjective, causing inter-observer variations even among senior pathologists. The subjectivity of the application of morphological criteria in visual classification motivates the use of computer-aided diagnosis (CAD) systems to improve the diagnosis accuracy, reduce human error, increase the level of interobserver agreement, and increased reproducibility.

There are many methods developed for digital pathology image analysis, from rule-based to applications of machine learning. Recently, deep learning-based approaches were shown to

outperform conventional machine learning methods in many image analysis tasks, automating end-to-end processing.

Previous deep learning-based applications in histological microscopic image analysis have demonstrated their potential to provide utility in diagnosing breast cancer. In this project, we present an approach for histology microscopy image analysis for breast cancer type classification.

Our approach utilizes DNNs for feature extraction and gradient boosted trees for classification and, to our knowledge, outperforms other similar solutions.

LITERATURE SURVEY

Breast cancer classification using a capsule network with preprocessed histology images written by Anupama M A, Sowmya V, Soman K P. In this paper they have classified breast cancer as diagnosed using histology images and have classified different types of breast cancer using histology images. This is done by effective image classification techniques. Among different image processing algorithms, deep learning gives the best performance for image classification applications. There are different convolutional neural network architectures used for classification purposes such as AlexNet, Inception-Net, ResNet, etc. Since conventional convolutional neural networks have a lot of drawbacks, the architecture used for the current study is capsule networks, which capture spatial and orientational information.

They proposed this work which shows that the accuracy of the Capsule Network model is improved due to the preprocessing of the histology images. The aim of the present work is to classify breast cancer using deep learning so that it can be used as an automated tool to assist doctors' diagnoses. In their proposed method, the histology images are fed as an input to the capsule net architecture. There are mainly 3 layers in the capsule net namely the input layer, hidden layer, and output layer. In the Input layer they used after applying the stain normalization, the extracted patches are resized to a 28×28 and the hidden layer of the convolution layer and capsule layer are the main components of the hidden layer. The kernel size of the filter used is 3 and it does the dot product between the filter and chunks of the input image. The next layer is the capsule layer of size 51. A nested set of neural layers is termed capsules. Then classification is done by a fully connected layer is the last layer of capsule net architecture. A fully connected layer connects every neuron in one layer to every neuron in another layer and softmax activation gives the probability distribution.

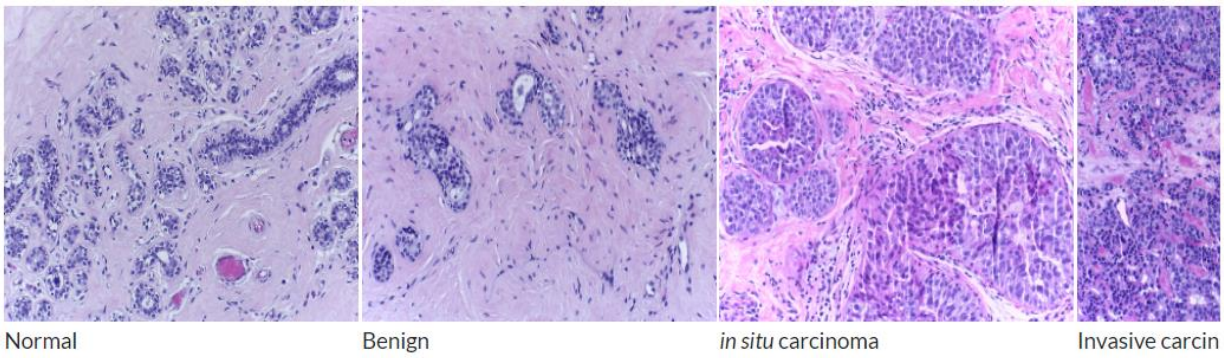
So their present work is based on the classification of breast cancer using capsule net architecture. From the work, it is clear that the performance of the conventional architectures can be improved by data pre-processing and parameter tuning. The results show that their method can be used as an automated tool to assist doctors in disease diagnosis, which may lead

to higher concentration in the treatment at early stages rather than diagnosis and can increase the cancer survival rate.

DATASET

This project uses the dataset provided by the International Conference on Image Analysis and Recognition [ICIAR] on Breast Cancer Histology Images. The dataset is composed of Hematoxylin and eosin (H&E) stained breast histology microscopy and whole-slide images.

The microscopy images are labeled as normal, benign, in situ carcinoma or invasive carcinoma according to the predominant cancer type in each image. The annotation was performed by two medical experts and images where there was disagreement were discarded.



The dataset contains a total of 400 microscopy images in ".tiff" format in RGB Colour (3 channels) in 2048x1536 pixels, distributed as follows:

Normal: 100
Benign: 100
in situ carcinoma: 100
Invasive carcinoma: 100

Each image takes about 10 MB – 20 MB of space and is labeled image-wise. Whole-slide images are high-resolution images containing the entire sampled tissue. In this sense, microscopy images are just details of the whole-slide images. Because of that, each whole-slide image can have multiple normal, benign, in situ carcinoma, and invasive carcinoma regions. Each image has a corresponding list of labeled coordinates that enclose benign, in situ carcinoma, and invasive carcinoma regions.

METHODOLOGY

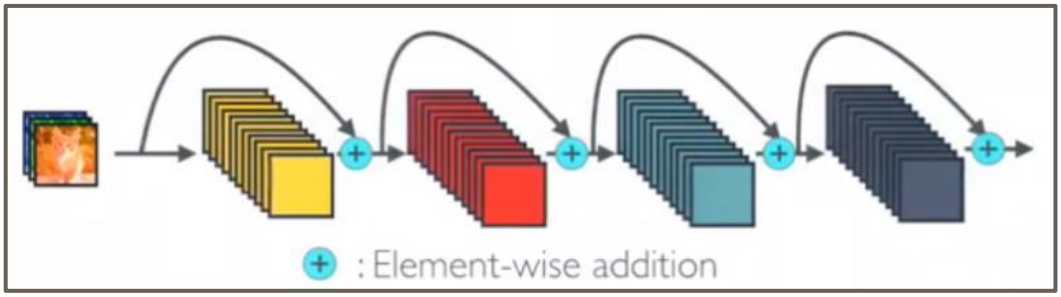
The images were first taken and preprocessed. For this we had to first read and load all the .tiff

lossless image files, each of size 10 - 20 MB and convert and store them as numpy arrays. This reduced all the images' sizes by a huge margin. Also they were resized to 299x299 size because it was the most optimal size for this work as we work with multiple models and this size suits all.

After loading the images in numpy vectorised format, it is then stored and this file is now accessed and passed into all the models used in this work and compared.

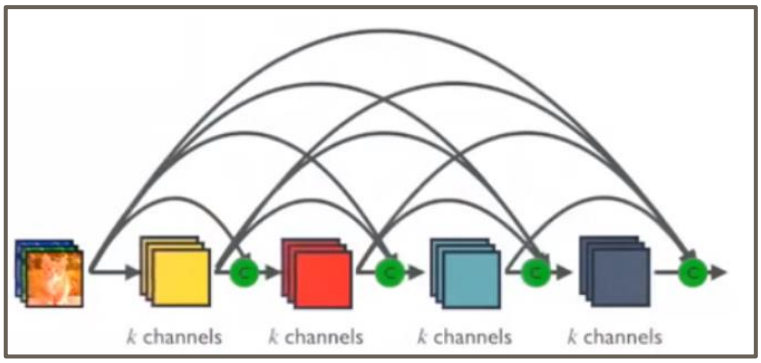
EXPERIMENTAL CODE 1

DenseNet: Deep CNNs encounter the problem of vanishing gradients. ResNets addressed this problem by including skip connections in its architecture, which in turn, aided in gradient propagation.



Identity element addition in ResNets

DenseNets adopt a similar strategy: make direct connections from the 1st layer's feature map to every other forthcoming feature map (i.e, the remaining 'L-1' feature maps); repeat this for the 2nd layer's feature map, and so on. Therefore, a network with 'L' layers will have (L + (L-1) + (L-2) + … + 2 + 1) = L(L+1)/2 connections amongst feature maps.



DenseNets have direct connections among feature maps (concatenation)

However, since any layer takes feature maps from all previous layers as input, there will be a huge accumulation of feature maps as the layers go deeper. This can be called, "the feature map explosion."

To avoid this, only a small fixed number of feature maps are given out of every layer. The whole architecture is constituted by groups of 3 dense layers. Such a group is called a *DenseNet block*. So, every dense block has a predefined number of layers & the predefined number of feature maps coming out of every layer.

The output from a dense block is given to a *transition layer*. The transition layer performs pointwise convolution followed by MaxPooling and keeps the output size in check.

The feature map coming out of every dense block has the same shape. Therefore, the output of every transition layer, i.e., the output after MaxPooling, will also be of the same shape. This allows the output of 1 transition layer to be directly connected to the outputs of forthcoming transition layers.

The following are a few significant points pertaining to DenseNets:

Lesser parameters: For every layer, the number of output feature maps is fixed. Therefore, only a fixed number of kernels need to be learned per layer.

Implicit supervision: Some architectures use auxiliary cost functions. Ex: Feature maps from intermediate layers of architecture can be passed to say, an SVM classifier, to perform classification. The corresponding error can then be directly back-propagated to that intermediate layer. Similarly, for DenseNets, since the feature maps from every layer are connected to the output, the error can directly be back-propagated to the intermediate layers. This pushes the intermediate layers to learn only features that are crucial for classification.

Growth rate: number of feature maps output by a layer within the Dense block.

Dense connectivity: Each layer inside a dense block takes as input the features from succeeding layers in the block.

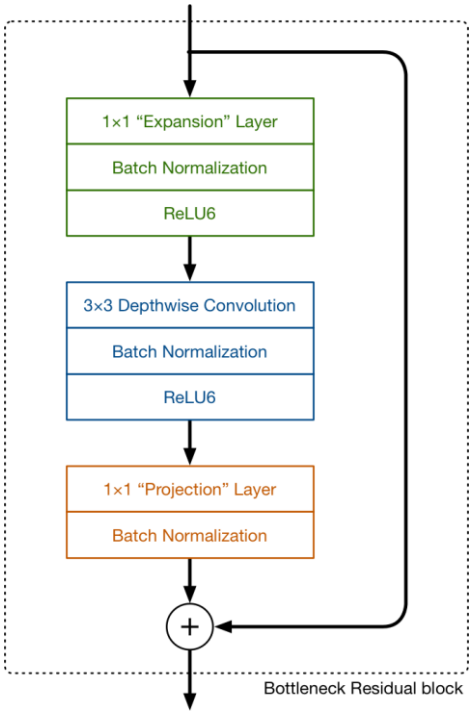
Composite functions: For every filter in a given layer, the following actions are implemented: Batch Normalization, ReLU activation, & convolution.

Owing to the aforementioned advantages, DenseNets achieve better performance even with a fewer number of parameters.

EXPERIMENTAL CODE 2

In MobileNetV2, there are two types of blocks. One is residual block with stride of 1. Another one is block with stride of 2 for downsizing. There are 3 layers for both types of blocks. This time, the first layer is 1×1 convolution with ReLU6. The second layer is the depthwise convolution. The third layer is another 1×1 convolution but without any non-linearity. It is claimed that if ReLU is used again, the deep networks only have the power of a linear classifier

on the non-zero volume part of the output domain. MobileNetV2 outperforms MobileNetV1 and ShuffleNet (1.5) with comparable model size and computational cost. With a width multiplier of 1.4, MobileNetV2 (1.4) outperforms ShuffleNet (×2), and NASNet with faster inference time.

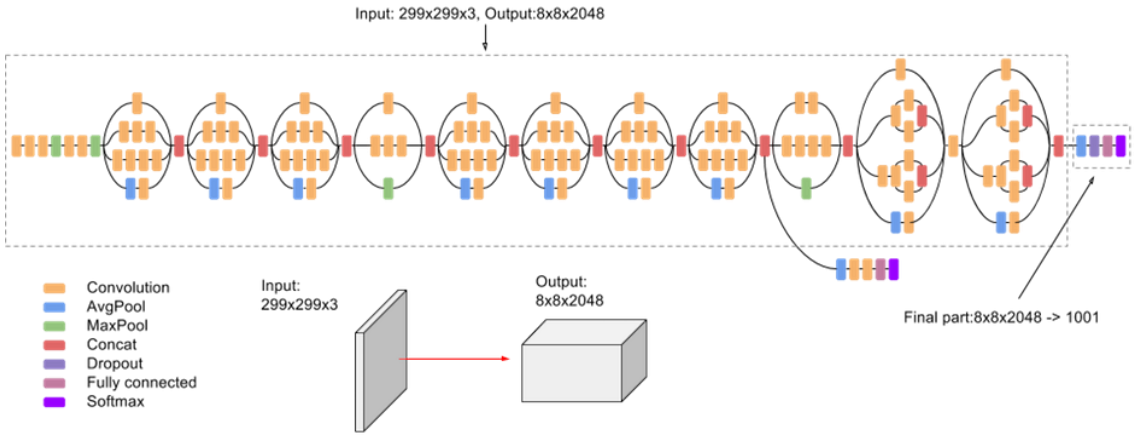


There are three convolutional layers in the block. The last two are the depthwise convolution that filters the inputs, followed by a 1×1 pointwise convolution layer. However, this 1×1 layer has a different job. In Mobile Net v1 the pointwise convolution either kept the number of channels the same or doubled them. In v2 it does the opposite: it makes the number of channels smaller. Hence this layer is now known as the projection layer — it projects data with a high number of dimensions (channels) into a tensor with a much lower number of dimensions.

EXPERIMENTAL CODE 3

Inception v3 is an image recognition model that has been shown to attain greater than 78.1% accuracy on the ImageNet dataset. The model is the culmination of many ideas developed by multiple researchers over the years. It is based on the original paper: "Rethinking the Inception Architecture for Computer Vision" by Szegedy, et. al.

The model itself is made up of symmetric and asymmetric building blocks, including convolutions, average pooling, max pooling, concatenations, dropouts, and fully connected layers. Batch normalization is used extensively throughout the model and applied to activation inputs. Loss is computed using Softmax.

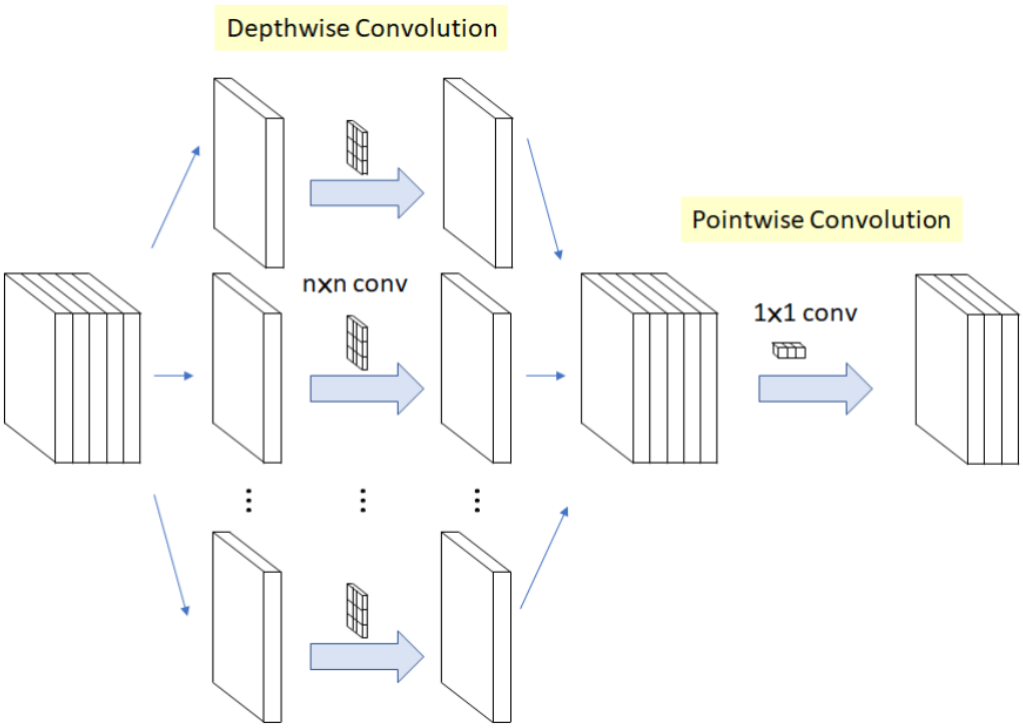


Inception Net v3 Architecture

EXPERIMENTAL CODE 4

Xception by Google which stands for Extreme version of Inception is reviewed. With a modified depthwise separable convolution, it is even better than Inception-v3.

1. Original Depthwise Separable Convolution

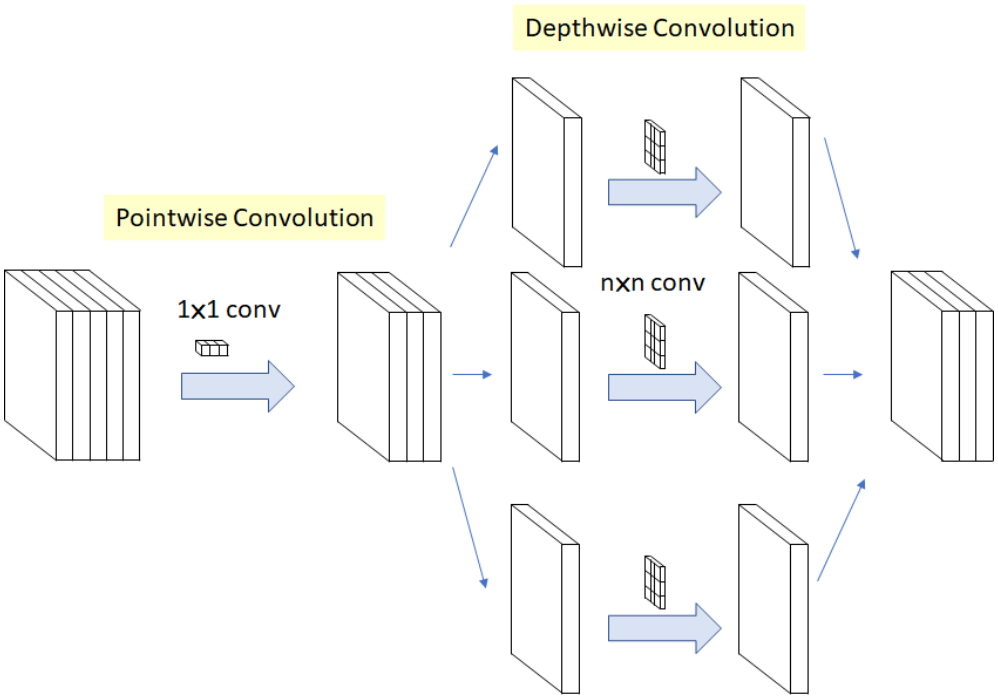


The original depthwise separable convolution is the depthwise convolution followed by a pointwise convolution.

1. Depthwise convolution is the channel-wise n×n spatial convolution. Suppose in the figure above, we have 5 channels, then we will have 5 n×n spatial convolutions.
2. Pointwise convolution actually is the 1×1 convolution to change the dimension.

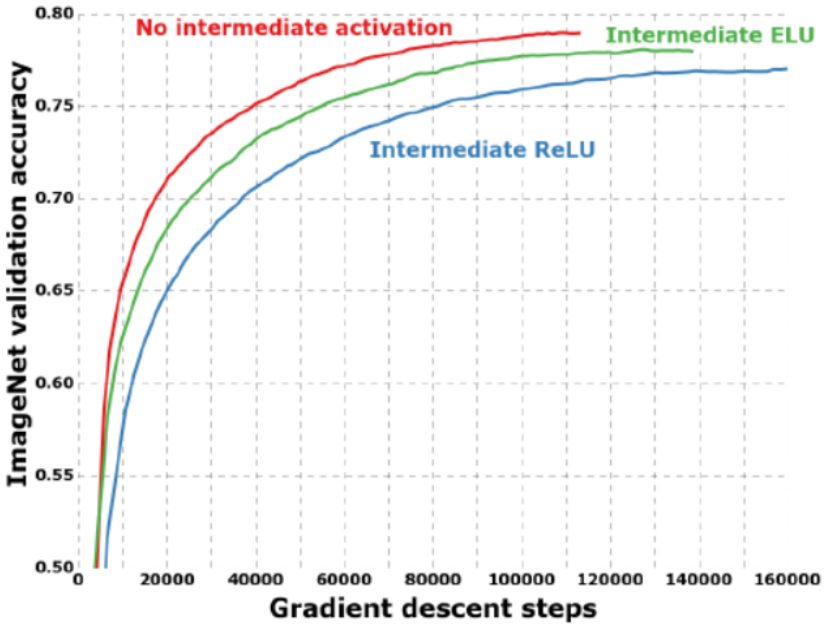
Compared with conventional convolution, we do not need to perform convolution across all channels. That means the number of connections is fewer and the model is lighter.

2. Modified Depthwise Separable Convolution in Xception



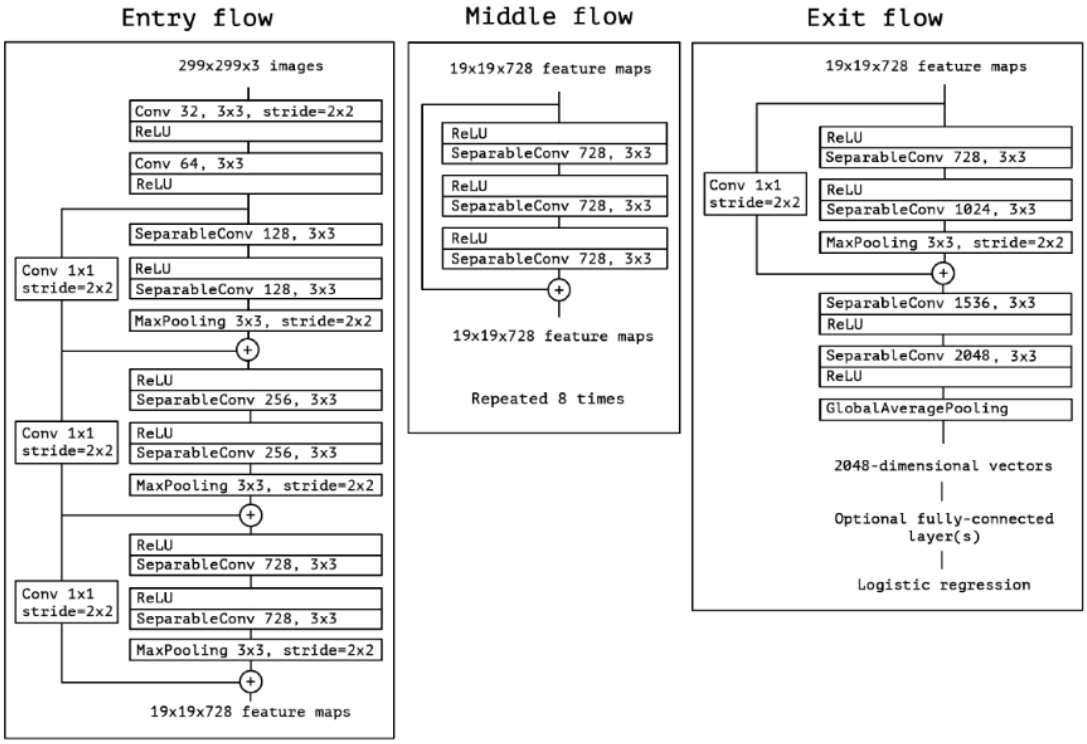
The modified depthwise separable convolution is the pointwise convolution followed by a depthwise convolution. This modification is motivated by the inception module in Inception-v3 that 1×1 convolution is done first before any n×n spatial convolutions. Thus, it is a bit different from the original one. (n=3 here since 3×3 spatial convolutions are used in Inception-v3.)

The modified depthwise separable convolution with different activation units is tested. As from the above figure, the Xception without any intermediate activation has the highest accuracy compared with the ones using either ELU or ReLU.



The order of operations: As mentioned, the original depthwise separable convolutions as usually implemented perform first channel-wise spatial convolution and then perform 1×1 convolution whereas the modified depthwise separable convolution performs 1×1 convolution first then channel-wise spatial convolution. This is claimed to be unimportant because when it is used in the stacked setting, there are only small differences that appeared at the beginning and at the end of all the chained inception modules. The Presence/Absence of Non-Linearity: In the original Inception Module, there is non-linearity after the first operation. In Xception, the modified depthwise separable convolution, there is NO intermediate ReLU non-linearity.

3. Overall Architecture



Overall Architecture of Xception (Entry Flow > Middle Flow > Exit Flow)

As in the figure above, SeparableConv is the modified depthwise separable convolution. We can see that Separable Convolutions are treated as Inception Modules and placed throughout the whole deep learning architecture. And there are residual (or shortcut/skip) connections, originally proposed by ResNet, placed for all flows.

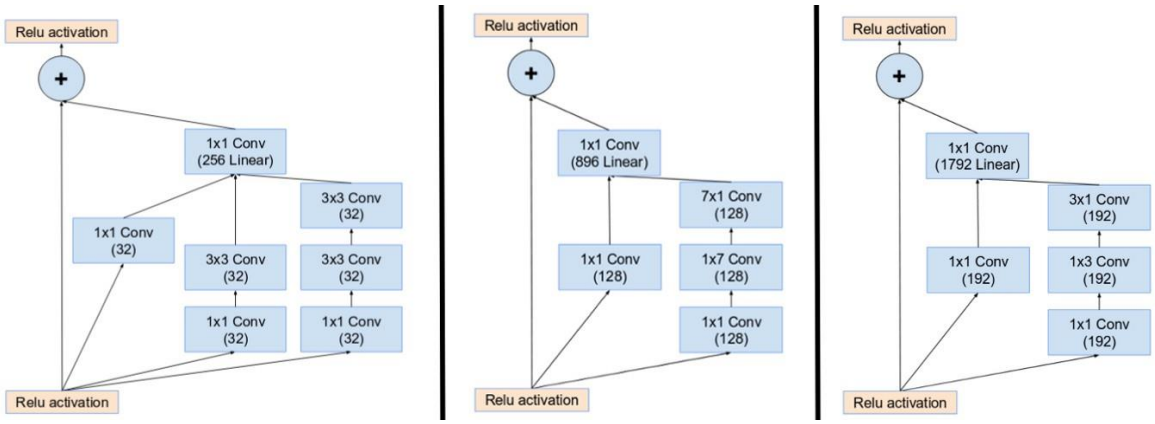
The Xception architecture has 36 convolutional layers forming the feature extraction base of the network. In our experimental evaluation, we will exclusively investigate image classification and therefore our convolutional base will be followed by a logistic regression layer. Optionally one may insert fully-connected layers before the logistic regression layer. The 36 convolutional layers are structured into 14 modules, all of which have linear residual connections around them, except for the first and last modules. In short, the Xception architecture is a linear stack of depthwise separable convolution layers with residual connections.

As seen in the architecture, there are residual connections. Here, it tests for Xception using the non-residual version. From the above figure, we can see that the accuracy is much higher when using residual connections. Thus, the residual connection is extremely important.

EXPERIMENTAL CODE 5

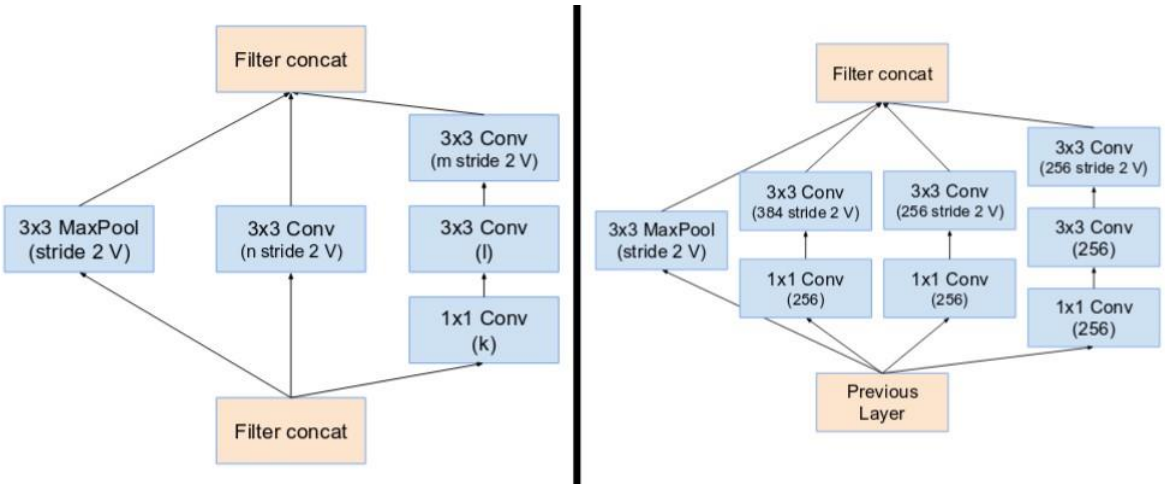
Inspired by the performance of the ResNet, a hybrid inception module was proposed. There are two sub-versions of Inception ResNet, namely v1 and v2. The differences between these two sub-versions are given below.

- Inception-ResNet v1 has a computational cost that is similar to that of Inception v3.
- Inception-ResNet v2 has a computational cost that is similar to that of Inception v4.
- They have different stems, as illustrated in the Inception v4 section.
- Both sub-versions have the same structure for the modules A, B, and C and the reduction blocks. The only difference is the hyper-parameter settings. In this section, we'll only focus on the structure. Refer to the paper for the exact hyper-parameter settings (The images are of Inception-Resnet v1).



Inception ResNetv2 blocks A, B, and C

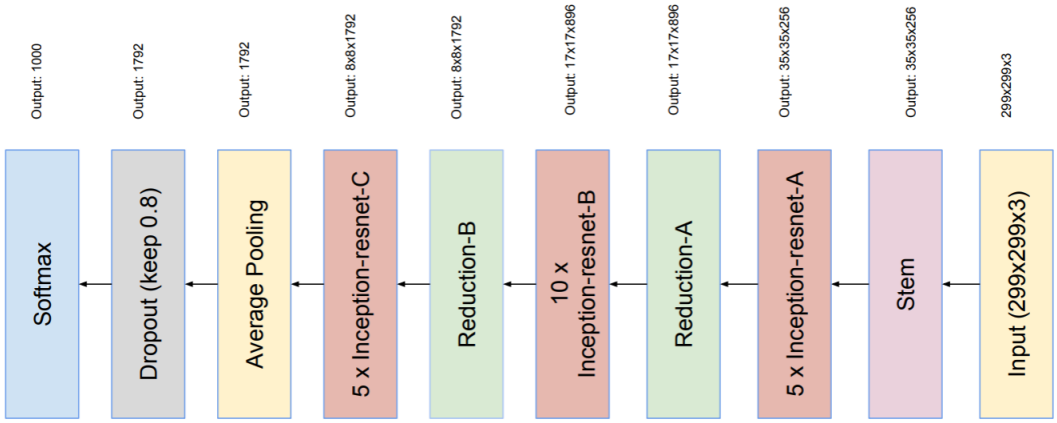
The main goal of this architecture is to introduce residual connections that add the output of the convolution operation of the inception module to the input.
For residual addition to work, the input and output after convolution must have the same dimensions. Hence, we use 1x1 convolutions after the original convolutions, to match the depth sizes (Depth is increased after convolution).



Reduction Blocks A and B

The pooling operation inside the main inception modules were replaced in favor of the residual connections. However, we can still find those operations in the reduction blocks. Reduction block A is the same as that of Inception v4. Networks with residual units deeper in the architecture caused the network to "die" if the number of filters exceeded 1000. Hence, to increase stability, the residual activations were scaled up, by a value around 0.1 to 0.3.

It was found that Inception-ResNet models were able to achieve higher accuracies at a lower epoch.



Inception ResNet v2 Architecture

RESULT AND DISCUSSION

The Breast Cancer data were successfully analyzed and the accuracy results from each model were evaluated. All models gave good accuracy scores. After training and analyzing all the

models we can safely come to a conclusion that mobile net v2 is the best model that gave the highest accuracy of them all. The following table shows the comparison of the model outputs.

MODEL NAME	TRAIN ACCURACY	TEST ACCURACY	VALIDATION ACCURACY
DENSE NET	97.92%	87.5%	81.88%
MOBILE NET v2	99.17%	90.62%	90.62%
INCEPTION NET v3	94.17%	70.63%	71.88%
XCEPTION NET	98.75%	92.5%	90.0%
INCEPTION ResNet v2	95.63%	86.25%	81.25%

Model Accuracy Scores Comparison

CONCLUSION

In this work five types of Deep Neural Net Algorithms were successfully implemented and the models classified the Breast Cancer Data with Good Accuracy values,, implemented with early stopping. The model that performed the best out of all five was Mobile Net v2 which had a training accuracy of 99.17% and a testing accuracy of 90.62%. Mobile Net v2 provides a very efficient mobile-oriented model that can be used as a base for many visual recognition tasks. Following this, the second-best model was Xception Net which had a training accuracy of 98.75% and a testing accuracy of 92.5%. This is mainly because of the separable convolutions done, it makes the model faster and more efficient than others.

REFERENCES

[1] Siegel R. L., Miller K. D., Jemal A., “Cancer statistics”, CA Cancer J Clin. 68:7- 30. W.-K. Chen, Linear Networks and Systems, Belmont, CA: Wadsworth, pp. 123–135, 2018.

[2] Sabour Sara, Frosst Nicholas, E. Hinton Geoffrey, “Dynamic Routing Between Capsules”, 31st Conference on Neural Information Processing Systems NIPS, 2017.

[3] Iesmantas T., Alzbutas R., “Convolutional Capsule Network for Classification of Breast Cancer Histology Images”, In Campilho A., Karray F., ter Haar Romeny B. (eds), Image Analysis and Recognition, Lecture Notes in Computer Science, vol 10882. Springer, Cham, ICIAR 2018.

[4] Ferreira C. A. et al. Classification of Breast Cancer Histology Images Through Transfer Learning Using a Pre-trained Inception Resnet V2. In: Campilho A., Karray F., ter Haar Romeny B. (eds)., Lecture Notes in Computer Science, vol 10882. Springer, Cham., Image Analysis and Recognition, ICIAR 2018.

[5] Esteva A., Kuprel B., Novoa R. A., Ko J., Swetter S. M., Blau H. M., Thrun S. “Dermatologist-level classification of skin cancer with deep neural networks”, Nature, vol.542, pp.115–118, 2017.

[6] Swapna G, Soman, K. P., and Vinayakumar R., “Automated detection of diabetes using CNN and CNN-LSTM network and heart rate signals”, Procedia Computer Science, vol. 132, pp. 1253-1262, 2018.

[7] Deepika J., Sowmya V., Soman K. P., Image Classification Using Convolutional Neural Networks. International Journal of Scientific and Engineering Research. 5. 1661-1668.10.14299/ijser.2014.06.002, 2014