TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PULCHOWK CAMPUS

THESIS NO.: 072/MSI/608

Thyroid Ultrasonography Image Classification Based on Fine-tuned Convolutional Neural Network

by

PANKAJ CHANDRA

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE IN INFORMATION AND COMMUNICATION ENGINEERING

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

NOVEMBER, 2019

“Thyroid Ultrasonography Image Classification Based on Fine-tuned Convolutional Neural Network”

by

Pankaj Chandra

(072/MSI/608)

Thesis Supervisor

Dr. Basanta Joshi

A thesis report submitted in partial fulfillment of the requirements for the degree of Master of Science in Information and Communication Engineering

Department of Electronics and Computer Engineering

Institute of Engineering, Pulchowk Campus

Tribhuvan University

Lalitpur, Nepal

November, 2019

COPYRIGHT©

TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PULCHOWK CAMPUS, PULCHOWK

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

The undersigned certify that they have read and recommended to the Department of Electronics and Computer Engineering for acceptance, a thesis entitled "Thyroid Ultrasonography Image Classification Based on Fine-tuned convolutional neural network", submitted by Pankaj Chandra in partial fulfillment of the requirement for the award of the degree of "Master of Science in Information and Communication Engineering".

……

Supervisor/Programme Coordinator, Dr. Basanta Joshi
Lecture,
Department of Electronics and Computer Engineering,
Institute of Engineering, Tribhuvan University

……

External Examiner: Er. Bijay Kumar Roy
Director,
Nepal Telecommunication Authority (NTA)
Kamaladi, Kathmandu, Nepal

Date: 22 November, 2019

DEPARTMENT ACCEPTANCE

The thesis entitled “Thyroid Ultrasonography Image Classification Based on Fine-tuned Convolutional Neural Network” submitted by Mr. Pankaj Chandra in partial fulfillment of the requirement for the award of the degree of “Master of Science in Information and Communication Engineering” has been accepted as a bonafide record of work independently carried out by him in the department.

……

Dr. Surendra Shrestha

Head of the Department

Department of Electronics and Computer Engineering,

Pulchowk Campus,

Institute of Engineering,

Tribhuvan University,

Nepal.

ACKNOWLEDGEMENT

I would like to acknowledge to Department of Electronics and Computer Engineering, Pulchowk Campus for providing opportunity to carry out thesis for the development of our career.

Foremost, I would like to sincere gratitude to my thesis supervisor Dr. Basanta Joshi for the continuous support of my thesis study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing this thesis.

Besides my Supervisor, I would like to convey my deep gratitude to my thesis committee, Prof. Dr. Sashidhar Ram Joshi, Prof. Dr. Subarna Shakya, Dr. Surendra Shrestha (HOD, Department of Electronics and Computer Engineering) Dr. Sanjeeb Prasad Panday, Dr. Diwakar Raj Pant, Dr. Nanda Bikram Adhikari, Dr. Ram Krishna Maharjan, Dr. Aman Shakya, Er. Daya Sagar Baral and Er. Babu Ram Dawadi for their encouragement, insightful comments and suggestion.

I would also like to thank Er. Gaurav Giri (SDE, Department of Hydrology and Meteorology), Er. Krishna Kumar Jha (Electronic Engineer, Ministry of Communication and Information Technology) and all my friends who shared their views and ideas and helped me during thesis work.

ABSTRACT

Most commonly found thyroid nodules are benign which is less harmful in comparison to malignant nodules. Number of techniques are available such as Ultrasonography imaging, percutaneous biopsy to determine whether a nodule is benign or malignant. However, these techniques require well experienced and senior radiologists. Only benignity and malignancy classification sometime result unnecessary surgery. Current Classification scheme, Thyroid Imaging Reporting and Data System (TIRADS) further classified the benign and malignant nodule which preclude biopsies required or not. The ensemble RetinaNet in conjunction with US image which improve nodule characterization and reduce biopsies. RetinaNet is promising technique as it is a simpler one-stage object detector which is fast and efficient. RetinaNet has been proven to perform conventional object detection tasks but has not been tested on detecting in Thyroid nodules. Here ensemble RetinaNet has been implemented which classified thyroid nodules based on TIRADS classes successfully. To validate its performance, the experimental setup has been constructed using the thyroid digital image database (TDID). In addition to training and testing on the same dataset, evaluation of model set up is done by pre-trained ImageNet dataset. The diagnostic performance of the ensemble network model was calculated on the basis of precision, recall and F1 value. The precision value of the aforementioned network obtained up to 94% while recall value obtained up to 96% and F1 score obtained up to 93%.

Keywords: Thyroid Digital Image Database, Thyroid Nodule, TIRADS, RetinaNet, Ultrasonography.

TABLE OF CONTENTS

COPYRIGHT© iii
DEPARTMENT ACCEPTANCE v
ACKNOWLEDGEMENT vi
ABSTRACT vii
LIST OF FIGURES x
LIST OF TABLES xi
LIST OF ABBREVIATIONS xii
CHAPTER 1 INTRODUCTION 1
1.1 Background and Motivation 1
1.2 Organization of the Thesis 2
1.3 Problem statement 3
1.4 Objectives 4
CHAPTER 2 LITERATURE REVIEW 5
CHAPTER 3 THEORETICAL BACKGROUND 9
3.1 Convolutional Neural Network (ConvNet or CNN) 9
3.1.1 CNN architecture 9
3.2 Neuron Activation Function 11
3.2.1 Sigmoid Function 11
3.2.2 Hyperbolic Tangent Function(tanh) 11
3.2.3 Rectified Liner Unit (ReLU) 12
3.3 Parameter Tuning 13
3.3.1 Learning Rate 13
3.3.2 Batch Size 14
3.3.3 Dropout Regularization 14
3.4 Dataset Processing 15

3.5 Tools 15
CHAPTER 4 METHODOLOGY 16
4.1 System Block Diagram 16
4.1.2 Data Collection 17
4.1.3 Fine Tuning 19
4.1.4 AlexNet 19
4.1.5 GoogleNet 21
4.1.6 RetinaNet 22
4.1.7 Softmax 26
4.1.8 Evaluation Metrics 27
CHAPTER 5 RESULT AND DISCUSSION 29
5.1 Suspicious Nodular Area Detection 29
5.2 Parameter Tuning (Hyperparameter for CNN) 30
5.3 Comparison with different Fine-tuned Ensemble Network Model 33
5.3 Thyroid Nodular Classification 34
5.4 Test Result 37
CHAPTER 6 CONCLUSION AND LIMITATION 40
6.1 Conclusion and Limitation 40
6.2 Future Works 40
REFERENCES 41

LIST OF FIGURES

Figure 1:A CNN sequence to classify image 9
Figure 2 The Output of Sigmoid Function as x varies 12
Figure 3 The Output of tanh Function as x varies 12
Figure 4 The output of ReLU as x varies 13
Figure 5 Dropout Regularization 14
Figure 6 Dataset include the validation set to prevent Overfitting during training 15
Figure 7:Thyroid Ultrasonography Image Classification Block Diagram 16
Figure 8 Ultrasonography Image of Thyroid of TDID dataset 18
Figure 9:AlexNet system block diagram 20
Figure 10 A Structure of single Inception Layer 21
Figure 11:The network architecture of RetinaNet 23
Figure 12 Resnet Block 24
Figure 13 Original Image of Thyroid Nodule (TIRADS5) 29
Figure 14 Suspicious Nodule area Detection using Image Thresholding 30
Figure 15 Accuracy (a) and Loss (b) plot on different learning rate 31
Figure 16 Accuracy(a) and Loss(b) plot on varying the Batch Size 32
Figure 17 Accuracy (a) and Loss(b) plot of different ensemble CNN model 33
Figure 18: Training(a) and Validation(a) Accuracy of the given Network Model 36
Figure 19: Training (a) and Validation (b) loss of the given model 37
Figure 20 Prediction of ultrasonography of thyroid test image by the given model 38
Figure 21 Confusion Matrix for Classification of Thyroid Nodules 38

LIST OF TABLES

Table 1 : Simulation Environment and parameter of AlexNet 20
Table 2: Simulation Environment and parameter of GoogleNet 21
Table 3 : Simulation Environment and parameter of ResNet 24
Table 4: ImageNet dataset classification by Ensemble Network.....34
Table 5: Classification Performance of the given Fine-tuned Network 39

LIST OF ABBREVIATIONS

ANN	Artificial Neural Network
CAD	Computer Aided Design
CNN	Convolution Neural Network
LBP	Local Binary Pattern
MC-CNN	Multitask Cascaded Convolution Neural Network
ReLU	Rectified Linear Unit
ROI	Region of Interest
SIFT	Scale Invariant Feature Transform
SR	Super Resolution
TDID	Thyroid Digital Image Database
TIRADS	Thyroid Imaging Reporting and Data System
US	Ultrasonography
USSR	Unsupervised Super-Resolution

CHAPTER 1 INTRODUCTION

1.1 Background and Motivation

A thyroid nodule is a lump that can develop in thyroid gland. It can be solid or filled with fluid. It can have a single nodule or a cluster of nodules. Thyroid nodules are relatively common and rarely cancerous. Recent study shows that thyroid nodules can be found in 68% of adults undergoing a thyroid ultrasound. Thyroid nodules increase with age and are present in almost 10% of the adult population. Most of solitary thyroid nodules are benign, and few of thyroid nodules are malignant.

Deep learning models such as Convolutional Neural Networks (CNN) has proved its efficiency in various learning tasks, including the image classification problems. However, training a deep convolutional neutron network from beginning requires enormous number of images while the medical images are usually more difficult to gather and more cumbersome to process due to their particularities. Lack of sufficient images will result in problems like over-fitting; thus, two possible solutions are transfer learning and data augmentation. Transfer learning adopts pre-trained deep learning models and then fine-tuning the parameter with existing images in purpose of adjusting the pre-trained model to fit the current classification problem. As for data augmentation, the classical methods for augmenting images such as cropping, rotation, flipping and rescaling. But unlike other images that can easily be labeled and recognized, medical images needs well trained physicians to classifier various type of diseases. Additionally, traditional way of augmenting image data risk eliminating the paramount region of the image by random cropping, such as the tumor in an ultrasound Images.

Convolution is the basis of CNN and it works by having a kernel to capture specific local patterns and gradually assemble layers of local patterns together to form more general patterns. For example, given an image of a thyroid ultrasonography, a convolution may first extract edges in the first layer, then use those edges to construct simple shapes in the second layer and then use these shapes to determine higher-level features, such as thyroid nodules. By using the Convolutional Neural Networks (CNN) architecture for generalization, essentially making an assumption: all specific local patterns in testing data are arranged by a similar rule as in training data.

1.2 Organization of the Thesis

This thesis implements the thyroid ultrasonography image classification such as benign nodules as TIRADS 2 and TIRADS 3 and malignant nodules as TIRADS 4a, 4b, 4c and TIRADS 5 based on fine-tuned ensemble convolutional neural network. The thesis report is organized and presented Chapter wise.

Chapter I introduces with some background and motivational introduction about the thyroid nodules and their types and how it effects on health with some description on problem statement and introduces the objective of this thesis.

Chapter II is regarding the Literature Review where includes the different research was done previously closely related to this thesis work. It also includes the different researcher approach and technology that used in their research and their outcomes and limitation discussed.

Chapter III is regarding the Theoretical Background of this Thesis including convolutional neural network and how it extracts feature from image, different architecture of CNN, Neuron activation function, parameter tuning for deep neural network, data processing and tools used in this thesis work.

Chapter IV Methodology includes the overall system implementation to fullfill the objective of this thesis work. It also includes the architecture and simulation parameter and environment for different CNN such AlexNet, GoogleNet and RetinaNet and Dataset collection, different approach that support the deep CNN such as fine-tuning, ensemble and data augmentation. In this section also describe the classifier which classified the ultrasonography image and evaluation metrics which gives the system appropriateness.

Chapter V gives the result of the experiment done in this research and the discussion of the result.

Chapter VI gives overall completion and conclusion of the thesis work and describe the limitation of the thesis work and recommend future work to further enhance the performance of the work.

1.3 Problem statement

Recently, many guidelines have been established for radiologists to evaluate thyroid nodules based on ultrasound characteristics. However, since ultrasonography is susceptible to echo disturbances and speckle noises, ultrasonography based thyroid nodule diagnosis still heavily relies on rich experiences and delicate skills of senior radiologists. Less experienced practitioners may potentially have high misdiagnosis rate due to their inability of accurately comprehending ultrasonography characteristics. Mis-diagnosis might consequently call for unnecessary biopsy and surgery, that would make patients have much more pressure and anxiety, and at the same time unavoidably increase medical expense. To effectively leverage the high-quality diagnosis experiences gained by senior radiologists, smart thyroid diagnosis Computer Aided Design (CAD) system is urgently needed. The benign nodules and the malignant nodules both have a wide variety of styles and layouts. The benign nodules have irregular shapes, smooth regions, and boundaries whereas malignant nodules have irregular shapes, coarse regions, and boundaries. Therefore, the thyroid nodules are hard to be directly recognized based on color and shape features.

It is difficult to use hand-crafted features for thyroid nodule images to detect benign and malignant due to factors such as nodule composition, echogenicity, shape and calcification of the affected part of patient and differences in imaging devices. To address these problems, several studies have leveraged a deep convolutional neural network that does not require hand-crafted features. Deep learning technique that implicitly perform feature extraction on image data with deeper networks, generally learns more sophisticated representations of the image data. Training deep learning to perform this kind of automated feature extraction typically comes with the onus of requiring large volumes of labeled training data. When such training corpora are available, deep learning are capable of achieving state-of-the-art performance in general object recognition. CNNs may be indirectly limited when used with highly variable image datasets with limited samples (e.g., thyroid nodule images): shallow deep learning may be too general and would not be able to capture the subtle differences between such images while deep network may become highly sensitive to subtle differences and would not be able to capture the general similarity between such images. A method for classifying the different classes of thyroid nodule ultrasonography images using an ensemble of different CNN architectures such as

AlexNet, GoogleNet and RetinaNet. Ensemble learning is a machine learning process in which better predictive performance is obtained by combining the results from multiple classification models into one high quality classifier. The model network resolves the challenges associated with using deep learning on multi-class classification problems with limited and unevenly distributed sample data by using ensemble network that have been pre-trained on a large collection of natural images (> 1 million) and fine-tuning (optimizing) them using a smaller thyroid US image dataset (thousands).

1.4 Objectives

- To classify the US Thyroid Nodule Images using ensemble One-Stage Classifier model RetinaNet.
- To analyze the performance of model on various classes of the Thyroid Nodule based on TIRADS.

CHAPTER 2 LITERATURE REVIEW

Image patch classification is an important task in many different medical imaging applications. Customized Convolutional Neural Networks (CNN) with shallow convolution layer to classify lung image patches with interstitial lung disease (ILD). While many feature descriptors have been proposed over the past years, they can be quite complicated and domain-specific. CNN framework can automatically and efficiently learn the intrinsic image features from lung image patches that are most suitable for the classification purpose. The same architecture can be generalized to perform other medical image or texture classification tasks. [1]

Artificial Neural Network (ANN) has been studied for many years to solve complex classification problems including image classification. The distinct advantage of neural network is that the algorithm could be generalized to solve different kinds of problems using similar designs. In image classification problems, the descriptiveness and discriminative power of features extracted are critical to achieve good classification performance. Feature extraction techniques commonly used in medical imaging include intensity histograms, filter-based features and the recently very popular scale-invariant feature transform (SIFT) and local binary patterns (LBP).[2]

The key challenge for automatically classifying the modality of a medical image is due to the visual characteristics of different modalities: some are visually distinct while others may have only subtle differences. This challenge is compounded by variations in the appearance of images based on the diseases depicted and a lack of sufficient training data for some modalities. A new method for classifying medical images that uses an ensemble of different convolutional neural network (CNN) architectures. CNNs are a state-of-the-art image classification technique that learns the optimal image features for a given classification task. We hypothesize that different CNN architectures learn different levels of semantic image representation and thus an ensemble of CNNs will enable higher quality features to be extracted. The fine-tuning process leverages the generic image features from natural images that are fundamental for all images and optimizes them for the variety of medical imaging modalities. These features are used to train numerous multi-class classifiers whose posterior probabilities are fused to predict the modalities of unseen images.[3]

Deep learning in conjunction with professional image characterization could improve nodule characterization and reduce benign biopsies. The extracted features using convolutional autoencoders, local binary patterns as well as histogram of oriented gradients descriptors in association with medical professional thyroid image characterization. The experiment showed the classifiers using these features can improve negative predictive value of thyroid nodule evaluation using ultrasound.[4]

The method of transfer learning is applied to classify the malignant and benign thyroid nodules based on their ultrasound images. The principal steps are preprocessing, data augmentation and classification by transfer learning. The preprocessing concentrates in extracting the region of interest (ROI). Two techniques of data augmentation are realized, the traditional ways of augmenting images and a small convolutional network. The best accuracy on the augmented dataset via convolutional network attains 93.75%, which exceeds the results of other two datasets and in the meanwhile outperforms other relevant methods.[5]

In clinical practice, senior doctors could pinpoint nodules by analyzing global context features, local geometry structure, and intensity changes, which would require rich clinical experience accumulated from hundreds and thousands of nodule case studies. To alleviate doctors' tremendous labor in the diagnosis procedure, advocate a machine learning approach to the detection and recognition. Developing a multi-task cascade convolution neural network framework (MC-CNN) to exploit the context information of thyroid nodules. It may be noted that, the framework is built upon a large number of clinically-confirmed thyroid ultrasound images with accurate and detailed ground truth labels. Other key advantages of our framework result from a multi-task cascade architecture, two stages of carefully-designed deep convolution networks in order to detect and recognize thyroid nodules in a pyramidal fashion, and capturing various intrinsic features in a global-to-local way.[6]

A novel unsupervised super-resolution (USSR) framework to solve the single image super-resolution (SR) problem in ultrasound images which lack of training examples. The powerful nonlinear mapping ability of convolutional neural networks (CNNs), without relying on prior training or any external data. We exploit the multi-scale contextual information extracted from the test image itself to train an image-specific network at test time. To capture valuable internal information, dilated convolution is

employed to increase the receptive field without increasing the network parameters. To speed up the convergence of the training, residual learning is used to directly learn the difference between the high-resolution and low-resolution images.[7]

Among the recent object detectors, RetinaNet is particularly promising as it is a simpler one-stage object detector that is fast and efficient while achieving state-of-the art performance. RetinaNet has been proven to perform conventional object detection tasks to validate its performance in diverse use cases, constructing several experimental setups using the public dataset INbreast and the in-house dataset GURO. In addition to training and testing on the same dataset (i.e. training and testing on INbreast) and evaluate mass detection model in setups using additional training data (i.e. training on INbreast + GURO and testing on INbreast). Also evaluate the model in setups using pre-trained weights (i.e. using Weights pre-trained on GURO, training and testing on INbreast). In all the experiments, the mass detection model achieves comparable or better performance than more complex state-of-the-art models including the two-stage object detector. Also, the results show that using the weights pre-trained on data sets achieves similar performance as directly using datasets in the training phase.[8]

Training a deep convolutional neural network (CNN) from scratch is difficult because it requires a large amount of labeled training data and a great deal of expertise to ensure proper convergence. A promising alternative to training from scratch is to fine-tune a CNN that has been pre-trained using, for instance, a large set of labeled natural images. The idea of fine-tuning is indeed attractive for medical imaging applications; however, the substantial differences between natural and medical images may compromise the effectiveness of such knowledge transfer [9]. Use of a pre-trained CNN with adequate fine-tuning outperformed or, in the worst case, performed as well as a CNN trained from scratch. The superiority of the fined-tuned CNNs became even more evident when reduced training sets were used for training and fine-tuning. The required level of fine-tuning differed from one application to another, neither shallow tuning nor deep tuning may be the optimal choice for a particular application. Layer wise fine-tuning may offer a practical way to reach the best performance for the application at hand based on the amount of available data. The performance of the CNN-based systems was greater than that of the handcrafted counterparts, further favoring the use of CNNs in medical imaging as a powerful alternative to handcrafted approaches.

Deeper neural networks are more difficult to train, a residual learning framework to ease the training of networks that are substantially deeper than those used previously. We explicitly reformulate the layers as learning residual functions with reference to the layer inputs, instead of learning unreferenced functions. A comprehensive empirical evidence showing that these residual networks are easier to optimize, and can gain accuracy from considerably increased depth [10].

The selection of parameters is one of the most important tasks in the training of a neural network. The choice of activation and loss functions is particularly relevant as the formulation of training procedures strongly depends on the pairing of these functions. Different combinations of these functions present the formulations of pairings of most common activation and loss functions. The impact of these formulations, including natural pairings, on both binary and multi-class classification in artificial and real-world datasets [11].

CHAPTER 3 THEORETICAL BACKGROUND

3.1 Convolutional Neural Network (ConvNet or CNN)

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, ConvNets have the ability to learn these filters/characteristics. The architecture of a ConvNet is analogous to that of the connectivity pattern of Neurons in the Human Brain and was inspired by the organization of the Visual Cortex. Individual neurons respond to stimuli only in a restricted region of the visual field known as the Receptive Field. A collection of such fields overlaps to cover the entire visual area.

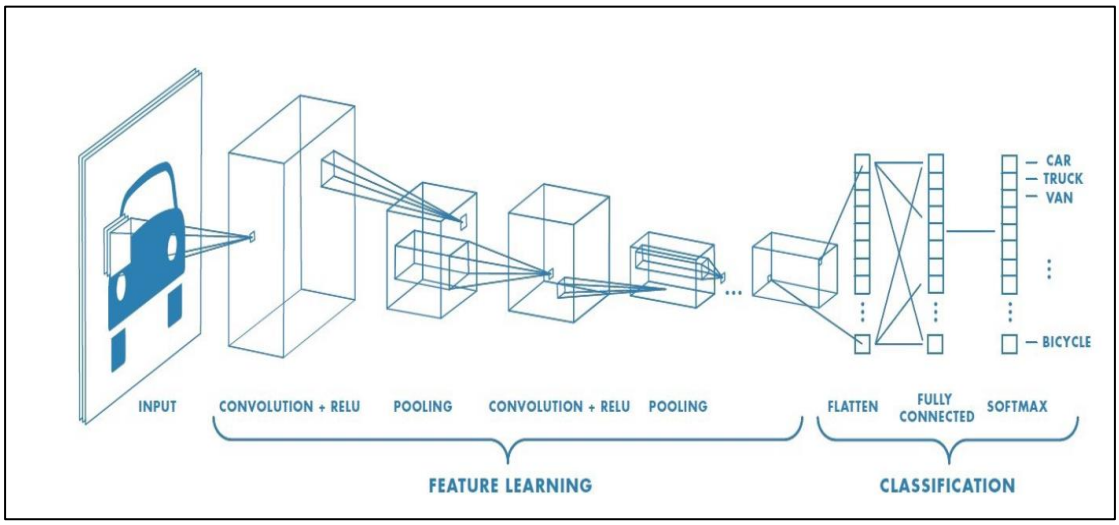


Figure 1:A CNN sequence to classify image

3.1.1 CNN architecture

A simple ConvNet is a sequence of layers, and every layer of a ConvNet transforms one volume of activations to another through a differentiable function. We use three main types of layers to build ConvNet architectures: Convolutional Layer, Pooling Layer, and Fully-Connected Layer.

The Convolution layer is the core building block of a Convolutional Network that does most of the computational heavy lifting. The CONV layer's parameters consist of a set of learnable filters. Every filter is small spatially (along width and height), but extends through the full depth of the input volume. For example, suppose that the input volume has size [32x32x3], (e.g. an RGB CIFAR-10 image). If the receptive field (or the filter size) is 5x5, then each neuron in the Conv Layer will have weights to a [5x5x3] region in the input volume, for a total of 5*5*3 = 75 weights (and +1 bias parameter). Notice that the extent of the connectivity along the depth axis must be 3, since this is the depth of the input volume.

Periodically insert a Pooling layer in-between successive Convolution layers in a ConvNet architecture. Its function is to progressively reduce the spatial size of the representation to reduce the number of parameters and computation in the network, and hence to also control overfitting. The Pooling Layer operates independently on every depth slice of the input and resizes it spatially.

Neurons in a fully connected layer have full connections to all activations in the previous layer, as seen in regular Neural Networks. Their activations can hence be computed with a matrix multiplication followed by a bias offset.

Convolutional layers are responsible for detecting certain local features in all locations of their input images. To detect local structures, each node in a convolutional layer is connected to only a small subset of spatially connected neurons in the input image channels. To enable the search for the same local feature throughout the input channels, the Thus, a convolutional layer with n kernels learns to detect n local features whose strength across the input images is visible in the resulting n feature maps. To reduce computational complexity and achieve a hierarchical set of image features, each sequence of convolution layers is followed by a pooling layer, a workflow reminiscent of simple and complex cells in the primary visual cortex. The max pooling layer reduces the size of feature maps by selecting the maximum feature response in overlapping or nonoverlapping local neighborhoods, discarding the exact location of such maximum responses. As a result, max pooling can further improve translation invariance. CNNs typically consist of several pairs of convolutional and pooling layers, followed by a number of consecutive fully connected layers, and finally a softmax layer, or regression layer, to generate the desired outputs. In more modern CNN architectures,

computational efficiency is achieved by replacing the pooling layer with a convolution layer with a stride larger than 1.

3.2 Neuron Activation Function

Activation functions are really important for an Artificial Neural Network to learn and make sense of something really complicated and Non-linear complex functional mappings between the inputs and response variable. They introduce non-linear properties to our Network. Their main purpose is to convert an input signal of a node in an ANN to an output signal. That output signal now is used as an input in the next layer in the stack. The most popular type of activation functions is described as:

3.2.1 Sigmoid Function

It is an activation function of form:

$$f(x) = \frac{1}{e^{-x}} \qquad (3.1)$$

It is easy to understand and apply but it has major reasons which have made it fall out of popularity.

- Vanishing gradient problem
- its output isn't zero centered. It makes the gradient updates go too far in different directions. 0 < output < 1, and it makes optimization harder.
- Sigmoid saturate and kill gradients
- Sigmoid have slow convergence.

3.2.2 Hyperbolic Tangent Function(tanh)

It is an activation function of form:

$$f(x) = \frac{2}{1 + e^{-2x}} - 1 \qquad (3.2)$$

Function output is zero centered and its value range in between -1 to 1 i.e. -1 < output < 1. Hence optimization is easier in this method. Deu to this reason in practice it is always preferred over Sigmoid function. But still it suffers from Vanishing gradient problem.

3.2.3 Rectified Liner Unit (ReLU)

It is an activation function of form:

$$f(x) = \max(0, x) \tag{3.3}$$

if x < 0, $f(x)$ = 0 and if x >= 0, $f(x)$= x.

It has become very popular in the past couple of years. It was recently proved that it had 6 times improvement in convergence from Tanh function. It avoids and rectifies vanishing gradient problem. So that almost all deep learning Models use ReLU nowadays.

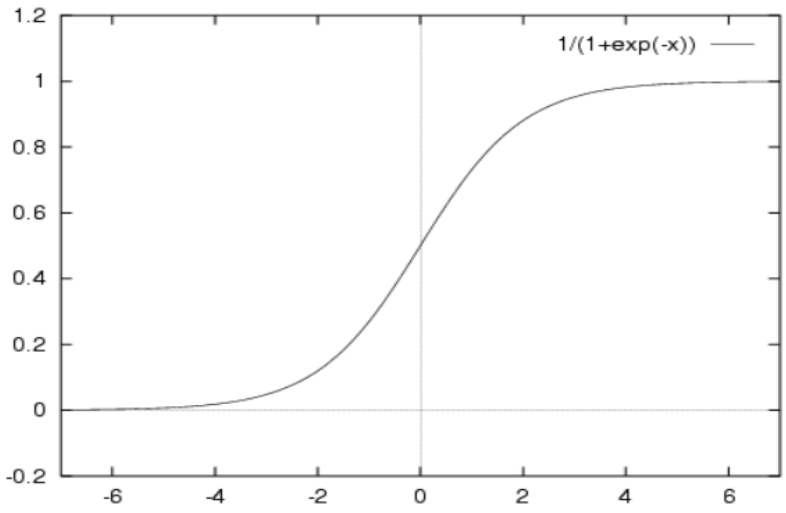


Figure 2 The Output of Sigmoid Function as x varies

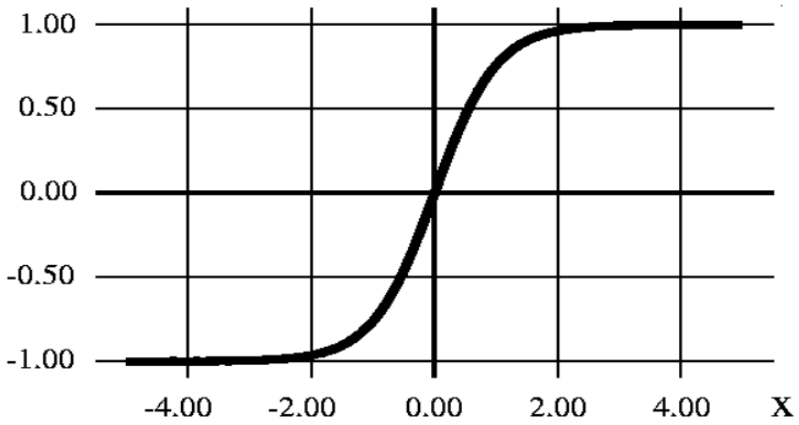


Figure 3 The Output of tanh Function as x varies

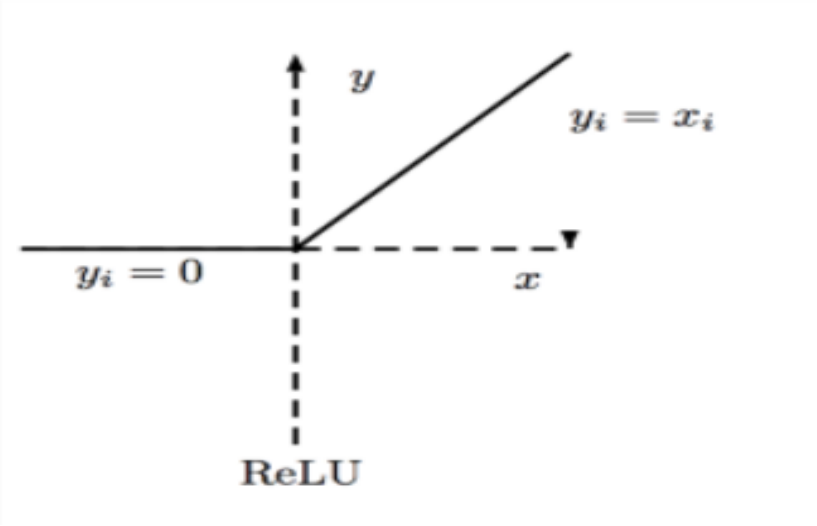


Figure 4 The output of ReLU as x varies

3.3 Parameter Tuning

The parameters of Neural Network that are fixed, also called hyperparameters, which are not learnt as part of the neural network, but rather passed as arguments to the classifier or regressor. Examples are the learning rate, optimizer or the kernel initializer that we set as part of building the neural network. The objective of hyperparameter optimization is to find the combination of hyperparameters that would result in an optimal model that would minimize the loss function. Loss function is the difference between the actual value and the predicted value.

3.3.1 Learning Rate

The learning rate is a hyperparameter that controls how much to change the model in response to the estimated error each time the model weights are updated. Choosing the learning rate is challenging as a value too small may result in a long training process that could get stuck, whereas a value too large may result in learning a sub-optimal set of weights too fast or an unstable training process. Deep learning neural networks are trained using the stochastic gradient descent algorithm. Stochastic gradient descent is an optimization algorithm that estimates the error gradient for the current state of the model using examples from the training dataset, then updates the weights of the model using the back-propagation of errors algorithm, referred to as simply backpropagation.

3.3.2 Batch Size

The number of examples from the training dataset used in the estimate of the error gradient is called the batch size and is an important hyperparameter that influences the dynamics of the learning algorithm. This involves using the current state of the model to make a prediction, comparing the prediction to the expected values, and using the difference as an estimate of the error gradient. This error gradient is then used to update the model weights and the process is repeated.

3.3.3 Dropout Regularization

The primary reason overfitting happens is because the model learns even the tiniest details present in the data. So, after learning all the possible patterns it can find, the model tends to perform extremely well on the training set but fails to produce good results on the validation and test sets. It falls apart when faced with previously unseen data. This neural network is overfitting on the training data. Suppose add a dropout of 0.5 to all these images. The model will randomly remove 50% of the units from each layer and we finally end up with a much simpler network:

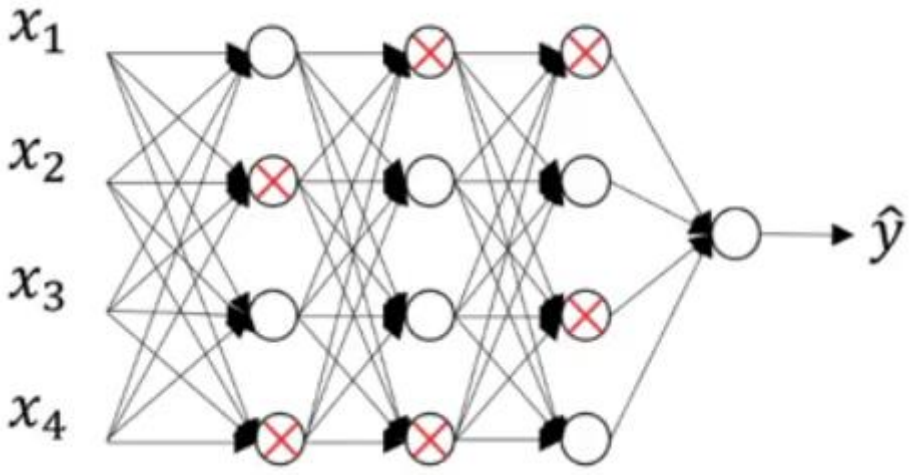


Figure 5 Dropout Regularization

The dropout regularization parameter set to a large value, the decay in the weights during gradient descent update will be more. Hence, the weights of most of the hidden units will be close to zero. Since the weights are negligible, the model will not learn much from these units. This will end up making the network simpler and thus reduce overfitting.

3.4 Dataset Processing

Training Dataset: The sample of data used to fit the model. The actual dataset that we use to train the model (weights and biases in the case of Neural Network). The model sees and learns from this data.

Validation Dataset: The sample of data used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyperparameters. The evaluation becomes more biased as skill on the validation dataset is incorporated into the model configuration. The validation set is used to evaluate a given model, but this is for frequent evaluation. Validation Dataset stop the training process as soon as overfitting start and prevent from poor generalization.

Test Dataset: The sample of data used to provide an unbiased evaluation of a final model fit on the training dataset. The Test dataset provides the gold standard used to evaluate the model. It is only used once a model is completely trained (using the train and validation sets).

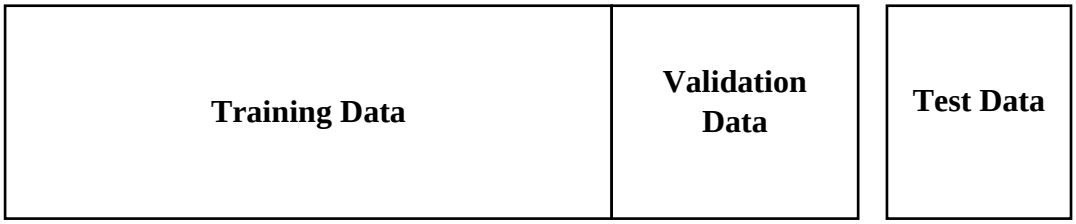


Figure 6 Dataset include the validation set to prevent Overfitting during training

3.5 Tools

Different computational task in this research are computed Using Python (Python 3.6.6), end to end open source platform for machine learning TensorFlow, TFLearn, TensorBord etc.

CHAPTER 4 METHODOLOGY

4.1 System Block Diagram

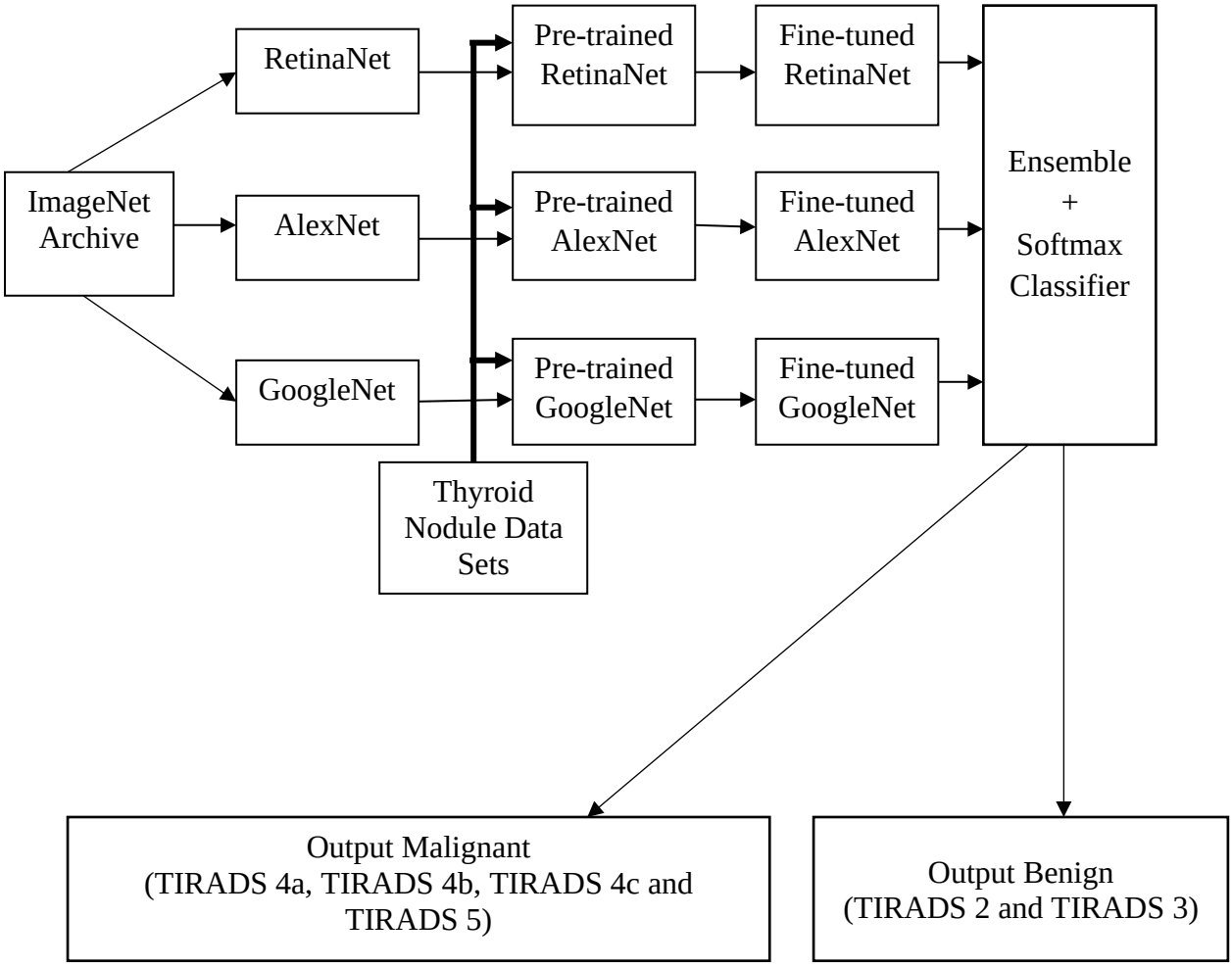


Figure 7:Thyroid Ultrasonography Image Classification Block Diagram

Figure7 shows an overall system block diagram of thyroid nodule image classification, in which first fine-tuned the CNN architectures that had been pretrained (initialized) on natural image dataset i.e. ImageNet. After that the dataset used for this research to fine tune the pretrained network comes from open access database for thyroid nodule TDID (Thyroid Digital Image Database). Each of the fine-tuned CNNs will then use a classifier generating softmax probabilities to determine the class of the image.

CNNs are trained with the back-propagation algorithm by minimizing the following cost function with respect to the unknown weights W which is given as

$$L = -\frac{1}{|X|}\sum_{i}^{|X|} \ln\left(p\left(y^i|X^i\right)\right) \qquad (4.1)$$

Where,

where $|X|$ denotes the number of training images, X^i denotes the i[th] training image with the corresponding label y^i, and $p\left(y^i|X^i\right)$ denotes the probability by which X^i is correctly classified.

Stochastic gradient descent is commonly used for minimizing this cost function, where the cost over the entire training set is approximated with the cost over mini-batches of data.

$$\gamma^t = \gamma^{\frac{tN}{|X|}}$$

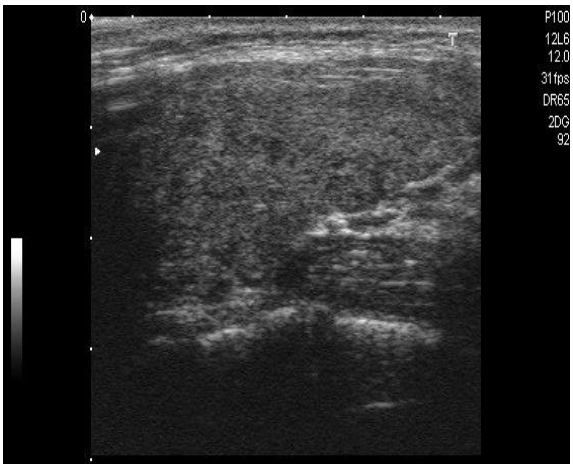
$$V_l^{t+1} = \mu V_l^t - \gamma^t \alpha_l \frac{\partial L}{W_l}$$

$$W_l^{t+1} = W_l^t + V_l^{t+1} \qquad (4.2)$$

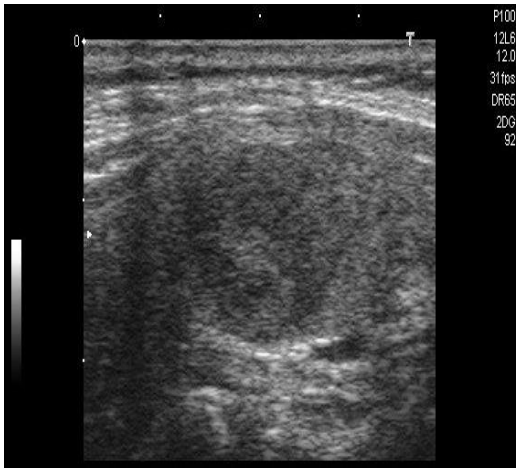
where α_l is the learning rate of the, l^{th} layer is the momentum that indicates the contribution of the previous weight update in the current iteration, and γ is the scheduling rate that decreases learning rate at the end of each epoch.

4.1.2 Data Collection

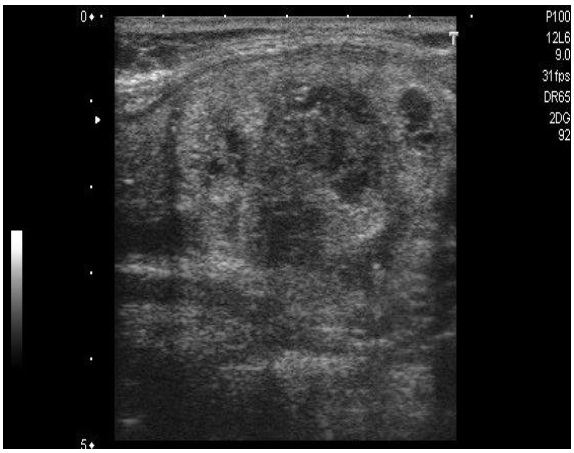
The dataset was collected from open access database for thyroid nodule TDID (Thyroid Digital Image Database), which contains in total 480 valid cases and the images in the grayscale. Among the 480 cases with TIRADS score, 280 cases were diagnosed as malignant (TIRADS score 4a, 4b, 4c and 5) and 200 cases as Benign (TIRADS score 2 and 3). The image augmentation process was used to produce 2000 number of datasets for training the convolutional Neural Network model. Among them 1400 images were used for training, 400 images used for validation and rest 200 images for test sets. The different classes of TIRADS images of TDID dataset is shown in figure 8.



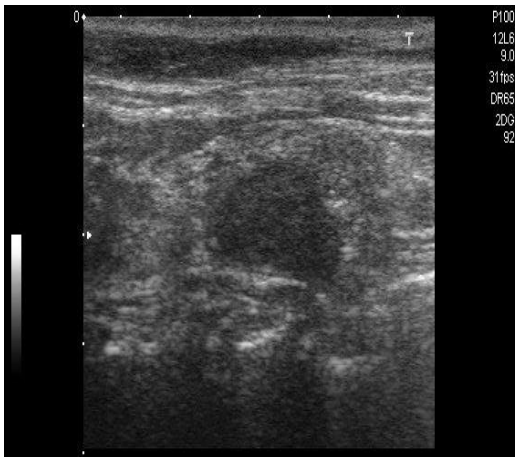
(a)TIRADS2



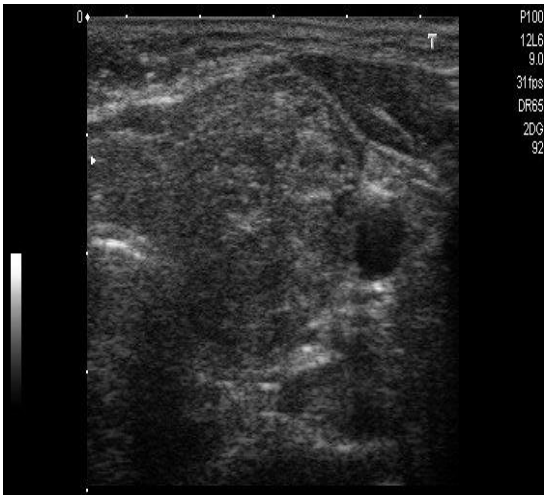
(b) TIRADS3



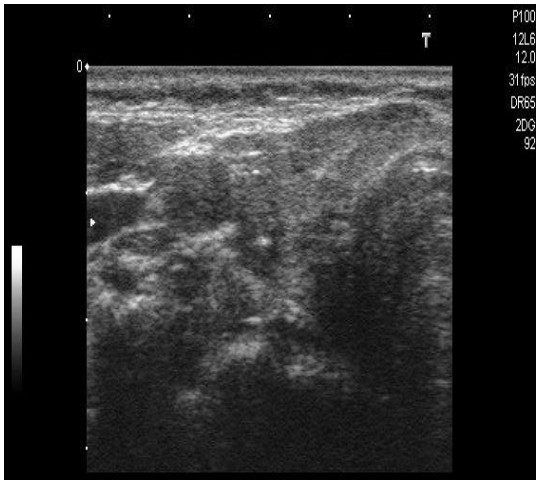
(c) TIRADS 4a



(d) TIRADS 4b



(e) TIRADS 4c



(f) TIRADS 5

Figure 8 Ultrasonography Image of Thyroid of TDID dataset

4.1.3 Fine Tuning

The iterative weight update in Eq 4.2 begins with a set of randomly initialized weights. Specifically, before the commencement of the training phase, weights in each convolutional layer of a CNN are initialized by values randomly sampled from a normal distribution with a zero mean and small standard deviation. However, considering the large number of weights in a CNN and the limited availability of labeled data, the iterative weight update, starting with a random weight initialization, may lead to an undesirable local minimum for the cost function. Alternatively, the weights of the convolutional layers can be initialized with the weights of a pre-trained CNN with the same architecture. The pre-trained net is generated with a massive set of labeled data from a different application. Training a CNN from a set of pre-trained weights is called finetuning and has been used successfully in several applications.

Fine-tuning begins with copying (transferring) the weights from a pre-trained network to the network we wish to train. The exception is the last fully connected layer whose number of nodes depends on the number of classes in the dataset. A common practice is to replace the last fully connected layer of the pre-trained CNN with a new fully connected layer that has many neurons as the number of classes in the new target application. In this research, it deals with 6-class classification tasks; therefore, the new fully connected layer has 6 neurons. After the weights of the last fully connected layer are initialized, the new network can be fine-tuned in a layer-wise manner, starting with tuning only the last layer, then tuning all layers in a CNN.

4.1.4 AlexNet

This well-established CNN follows standard neural network architecture of stacked and connected layers. It comprises eight layers that need to be trained, five convolutional layers followed by three fully connected layers, as well as max-pooling layers. first, second, and fifth convolutional layers are followed by overlapping max-pooling layers that make it more difficult for the network to overfit. The output of the fifth convolutional layer (after max-pooling) is fed into the stack of fully-connected layers. A rectified linear unit (ReLU) non-linearity is applied to each convolutional and fully connected layer to enable faster training.

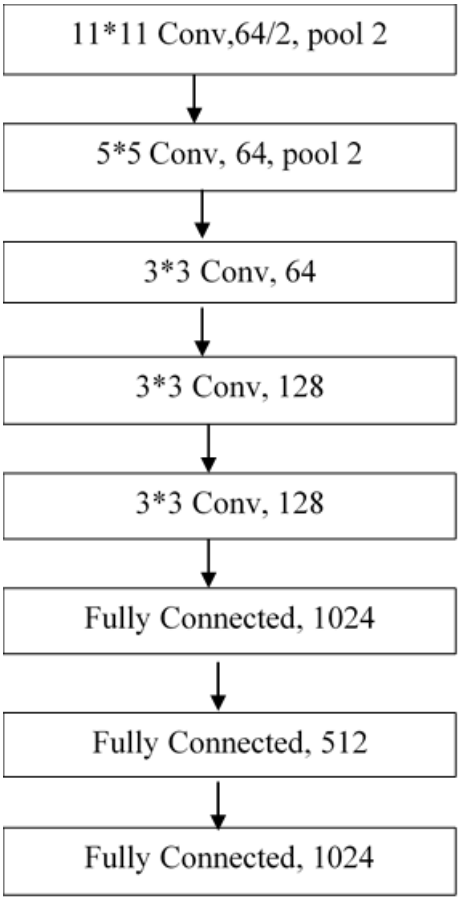


Figure 9:AlexNet system block diagram

Table 1 : Simulation Environment and parameter of AlexNet

Layer	**Width**	**Hight**	**Depth**	**Filter**	**Strides**
Input	300	300			
Conv1, ReLU	300	300	64	11*11	2
Max pool1	55	55	64	3*3	1
Conv2, ReLU	27	27	64	5*5	1
Maxpool2	27	27	32	3*3	1
Conv3, ReLU	27	27	64	3*3	1
Conv4, ReLU	13	13	128	3*3	1
Conv5, ReLU	6	6	128	3*3	1
Maxpool3	6	6	64		
Dropout (.5)					
Fully Connected, 1024					
Dropout (0.5)					

Fully connected, 512					
Regression, 6 classes, Softmax					

4.1.5 GoogleNet

This CNN architecture introduced a new "Inception" module, a subnetwork comprising of parallel convolutional filters whose outputs are concatenated. The repetition of the Inception modules captures the optimal sparse representation of the image while simultaneously reducing dimensionality. The network comprises 22 layers that require training (or 27 if pooling layers are also considered). Experiments have shown that GoogLeNet has fewer trainable weights than AlexNet and is more accurate.

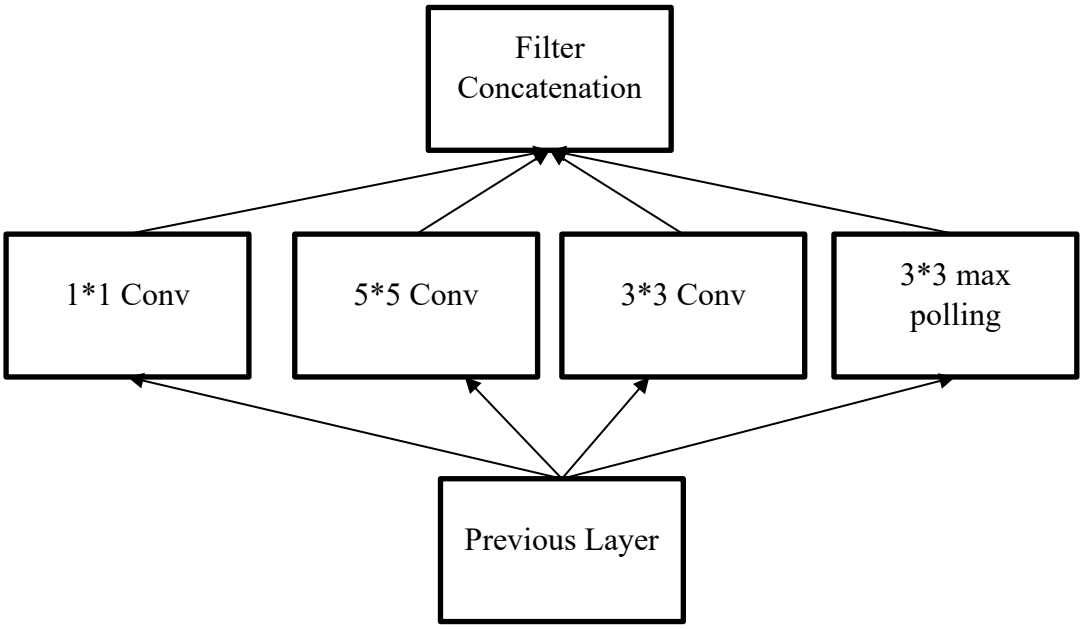


Figure 10 A Structure of single Inception Layer

Table 2: Simulation Environment and parameter of GoogleNet

Layer	**Width**	**Hight**	**Depth**	**Filter**	**Strides**
Input	300	300			
Conv1, ReLU	112	112	64	7*7	2
Max pool1	56	56	64	3*3	2
Conv2, ReLU	56	56	128	3*3	1

Maxpool2	28	28	128	3*3	2
Inception (3a), ReLU	28	28	256		
Inception (3b), ReLU	28	28	128		
Maxpool3	28	28	128	3*3	2
Inception (4a), ReLU	14	14	64		
Inception (4b), ReLU	14	14	64		
Inception (4c), ReLU	14	14	128		
Inception (4d), ReLU	14	14	256		
Inception (4e), ReLU	14	14	256		
Maxpool4	7	7	128	3*3	2
Inception (5a), ReLU	7	7	512		
Inception (5a), ReLU	7	7	512		
Average pool	1	1	512	7*7	1
Dropout(0.5)					
Fully Connected, 1024					
Dropout(0.4)					
Fully Connected, 512					
Regression, 6 classes, Softmax					

4.1.6 RetinaNet

RetinaNet is a single, unified network composed of a backbone network and two task-specific subnetworks. The backbone is responsible for computing a conv feature map over an entire input image and is an off-the-self convolution network. The first subnet performs classification on the backbones output; the second subnet performs convolution bounding box regression.

Backbone: Feature Pyramid network built on top of ResNet32 which can use classify the US Image.

Classification subnet: It predicts the probability of object presence at each spatial position for the object classes. Takes a input feature map with C channels from a pyramid level, the subnet applies four 3x3 conv layers, each with C filters and each

followed by ReLU activations. Finally, sigmoid activations are attached to the outputs. Focal loss is applied as the loss function.

Figure11 shows the overall architecture of RetinaNet. The backbone network computes convolutional feature map of an entire input image. The first subnetwork is the class subnet which classifies the output of the backbone network and the second subnetwork is the box subnet that performs convolutional bounding box regression. The architecture of RetinaNet is simpler than that of a two-stage object detector that is composed of independent multiple networks for classification and Regression.

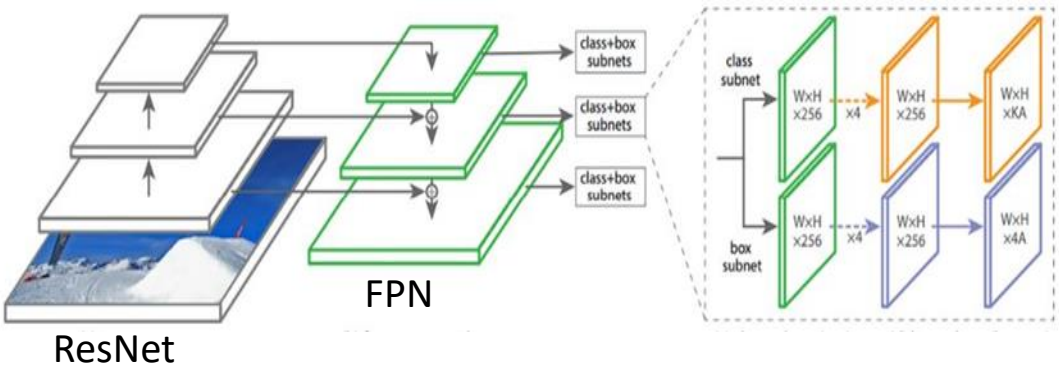


Figure 11:The network architecture of RetinaNet

4.1.6.1 ResNet

The degradation (of training accuracy) indicates that not all systems are similarly easy to optimize. Let us consider a shallower architecture and its deeper counterpart that adds more layers onto it. There exists a solution by construction to the deeper model: the added layers are identity mapping, and the other layers are copied from the learned shallower model. The existence of this constructed solution indicates that a deeper model should produce no higher training error than its shallower counterpart. But experiments show that our current solvers on hand are unable to find solutions that are comparably good or better than the constructed solution.

The residual block has two 3×3 convolutional layers with the same number of output channels. Each convolutional layer is followed by a batch normalization layer and a ReLU activation function. Then skip these two convolution operations and add the input directly before the final ReLU activation function. This kind of design requires that the

output of the two convolutional layers be of the same shape as the input, so that they can be added together.

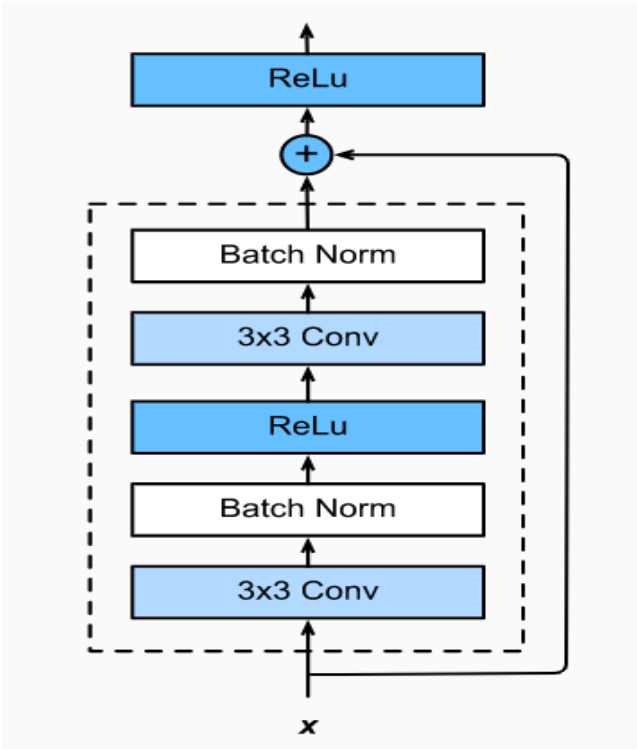


Figure 12 Resnet Block

Table 3 : Simulation Environment and parameter of ResNet

Layer	**Depth**	**Filter**	**Strides**
Input	1	N/A	1
Conv1	64	3*3	2
Conv2	64	3*3	2
Add Residual, ReLU			1
Conv3	128	3*3	2
Conv4	256	3*3	
Add Residual, ReLU	128		
Conv5	128	3*3	1
Conv6	64	3*3	
Add Residual, ReLU			

Conv7	128	3*3	
Conv8	256	3*3	
Add Residual, ReLU			
Conv9	128	3*3	2
Conv10	512	3*3	
Add Residual, ReLU			
…	…	…..	
…	…	….	
…	…	…..	
Conv31	512	3*3	1
Conv32	512	3*3	
Average Pool			
Fully connected, 1056			
Regression, 6 classes, Softmax			

4.1.6.2 Focal Loss

One-stage detectors that are applied over a regular, dense sampling of possible object locations have the potential to be faster and simpler, but have trailed the accuracy of two-stage detectors because of extreme class imbalance encountered during training Focal loss is the reshaping of cross entropy loss such that it down-weights the loss assigned to well-classified examples. The novel focal loss focuses training on a sparse set of hard examples and prevents the vast number of easy negatives from overwhelming the detector during training.

Focal loss function is simple extension of cross entropy (CE) loss function. CE loss function is defined as when the estimated probability for binary classification is defined as:

$$P_t = \begin{cases} P, & if\ y=1 \\ 1-P & otherwise \end{cases} \quad (4.3)$$

$$CE(P_t) = -\log(P_t) \quad (4.4)$$

$$FL(P_t) = -\alpha(1-)^{\gamma}\log(P_t) \quad (4.5)$$

The main property of CE loss function is that even samples that are easy to classify have a considerable amount of loss. Using CE loss function guarantees successful result when training a model on a balanced set.

4.1.7 Softmax

The softmax function is a generalization of the logistic function that highlights the largest values in a vector while suppressing those that are significantly below the maximum. When applied to a D-dimensional feature vector, the softmax function can be used as a non-linear variant of multinomial logistic regression to generate a vector of D probability values, the d-th element of which is the likelihood that the vector represents a member of the d-th class. The softmax function is widely used as the classification layer of many CNN architectures.

Oftentimes, we want our output vector to be a probability distribution over a set of mutually exclusive labels. For example, let's say we want to build a neural network to recognize handwritten digits from the MNIST dataset. Each label (0 through 9) is mutually exclusive, but it's unlikely that we will be able to recognize digits with 100% confidence. Using a probability distribution gives us a better idea of how confident we are in our predictions. As a result, the desired output vector is of the form below, where

$$\sum_{i=0}^{5} P_I = 1\,; \qquad (4.6)$$

$$[P_0 \quad P_1 \quad P_2 \ldots\ldots\ldots\ldots P_5]$$

This is achieved by using a special output layer called a softmax layer. Unlike in other kinds of layers, the output of a neuron in a softmax layer depends on the outputs of all the other neurons in its layer. This is because we require the sum of all the outputs to be equal to 1. Letting z_i be the logit of the i^{th} softmax neuron, we can achieve this normalization by setting its output to:

$$y_i = \frac{e^{z_i}}{\sum_j e^{z_j}} \qquad (4.7)$$

A strong prediction would have a single entry in the vector close to 1, while the remaining entries were close to 0. A weak prediction would have multiple possible labels that are more or less equally likely

4.1.8 Evaluation Metrics

4.1.8.1 Confusion Matrix

The Confusion matrix is one of the most intuitive and metrics used for finding the correctness and accuracy of the model. It is used for Classification problem where the output can be of two or more types of classes. the confusion matrix, is a table with two dimensions Actual and Predicted), and sets of classes in both dimensions. Actual classifications are columns and Predicted ones are Rows.

$$Confusion\ Matrix = \begin{bmatrix} TP & FN \\ FP & TN \end{bmatrix}$$

1. **True Positives (TP):** True positives are the cases when the actual class of the data point was 1(True) and the predicted is also 1(True).
2. **True Negatives (TN):** True negatives are the cases when the actual class of the data point was 0(False) and the predicted is also 0(False)
3. **False Positives (FP):** False positives are the cases when the actual class of the data point was 0(False) and the predicted is 1(True). False is because the model has predicted incorrectly and positive because the class predicted was a positive one. (1)
4. **False Negatives (FN):** False negatives are the cases when the actual class of the data point was 1(True) and the predicted is 0(False). False is because the model has predicted incorrectly and negative because the class predicted was a negative one. (0)
5. **Precision:** It is defined as:

   Precision = $\frac{TP}{TP+FP}$ (4.8)

   Which gives proportion of positive identifications was actually correct.

6. **Recall:** It is defined as:

Recall $= \frac{TP}{TP+FN}$ (4.9)

Which gives proportion of actual positives was identified correctly.

7. **F1-Score:** It is the harmonic mean of precision and recall.

   Recall $= 2 * \frac{Precesion*Recall}{Precesion+Recall}$ (4.10)

   Which gives the score which shows that how much the model is appropriate.

8. **Multiple Classification:** For multiple classification, $TP_{i,}$ $FP_{i,}$ and $FN_{i,}$ to respectively indicate true positives, false positives, and false negatives in the confusion matrix associated with the *i-th* class. Moreover, let precision be indicated by P and recall by R and are calculated as:

$$P = \frac{\sum_i TP_i}{\sum_i (TP_i + FP_i)} \quad (4.11)$$

$$R = \frac{\sum_i TP_i}{\sum_i (TP_i + FN_i)} \quad (4.12)$$

CHAPTER 5 RESULT AND DISCUSSION

5.1 Suspicious Nodular Area Detection

The method utilizes two bounds: horizontal projection and vertical projection, to locate the suspicious thyroid regions. This restricts the location of the segmentation and excludes some artifacts of the images. An example is shown in Figure12. Anatomical information in the image is obtained and the image is divided into three parts: skin, thyroid area, and dark region. Among of three parts, skin has higher gray levels, and the intensity of thyroid area is between skin and dark region. According to the characteristics, a horizontal projection is utilized to separate the thyroid area from other parts. We calculate the average intensity for each row of the image. Then Otsu's Thresholding were used which sets the average threshold value and separate the intensity information for both skin parts and nodules parts. Finally, the thresholding omitted the high intensity part and the average intensity part shown by a dark spot which indicated the nodular area in the given ultrasonography image. The dark spot in figure14 shows the thyroid affected area,

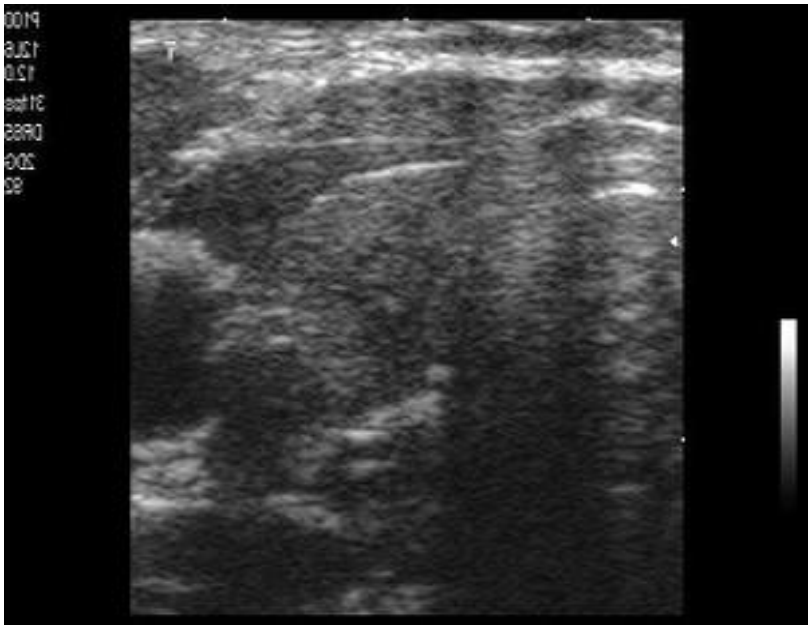


Figure 13 Original Image of Thyroid Nodule (TIRADS5)

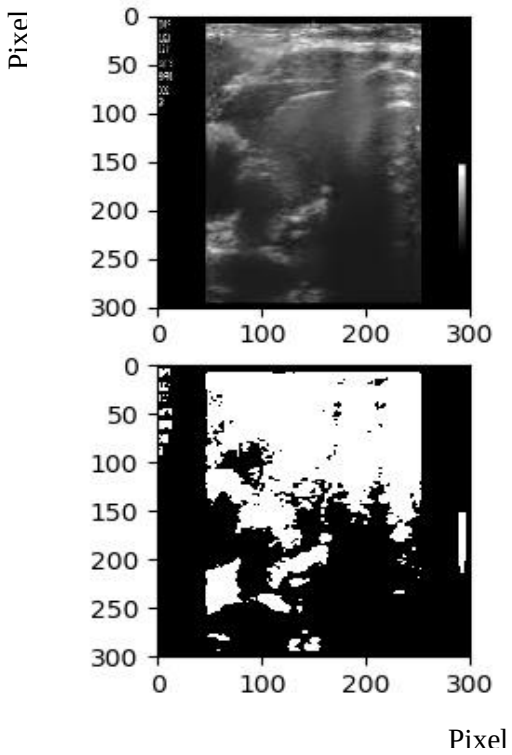
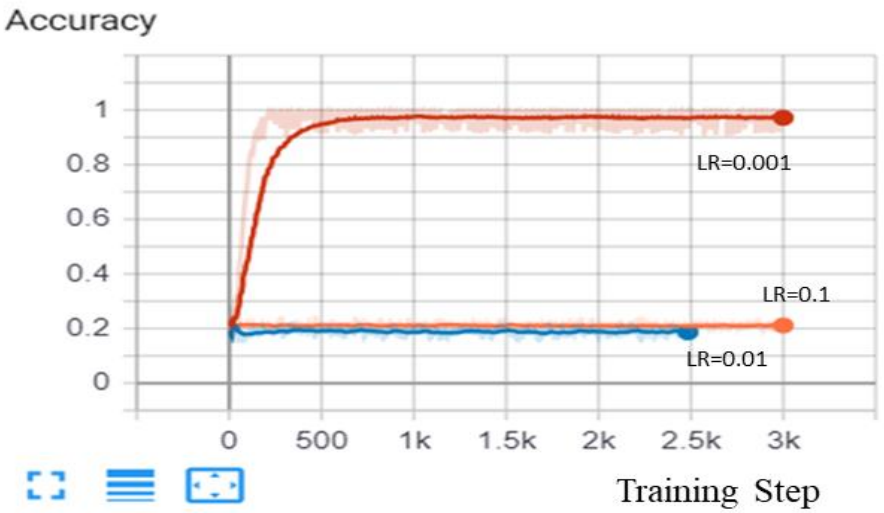


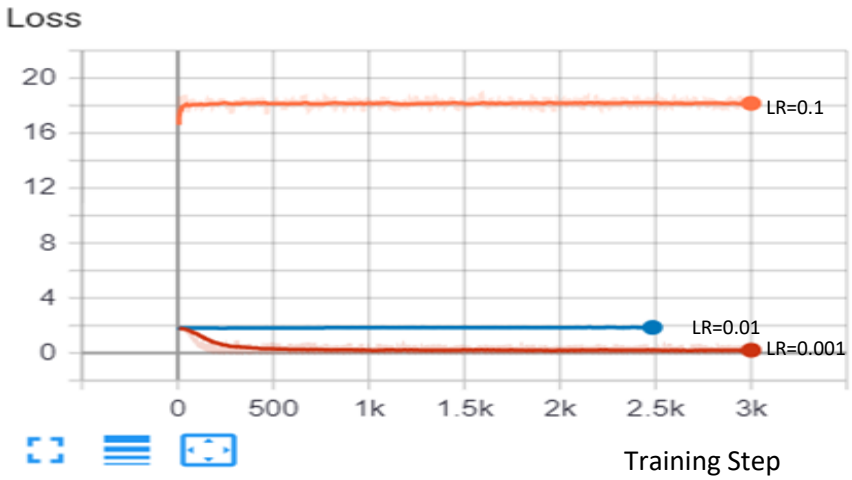
Figure 14 Suspicious Nodule area Detection using Image Thresholding

5.2 Parameter Tuning (Hyperparameter for CNN)

a. Learning Rate: An aforementioned network model i.e. ensemble of AlexNet, GoogleNet and RetinaNet was trained through the ultrasonography Image of training dataset by varying a learning rate as 0.1, 0.01 and 0.001. The accuracy and loss value at every training step was recorded in a log file and then all values plotted as shown in figure15. Figure15 shows that training with 0.1 and 0.01 learning rate did not converge and result low training accuracy and high validation losses whereas training with learning rate 0.001 results high training accuracy and low validation losses.



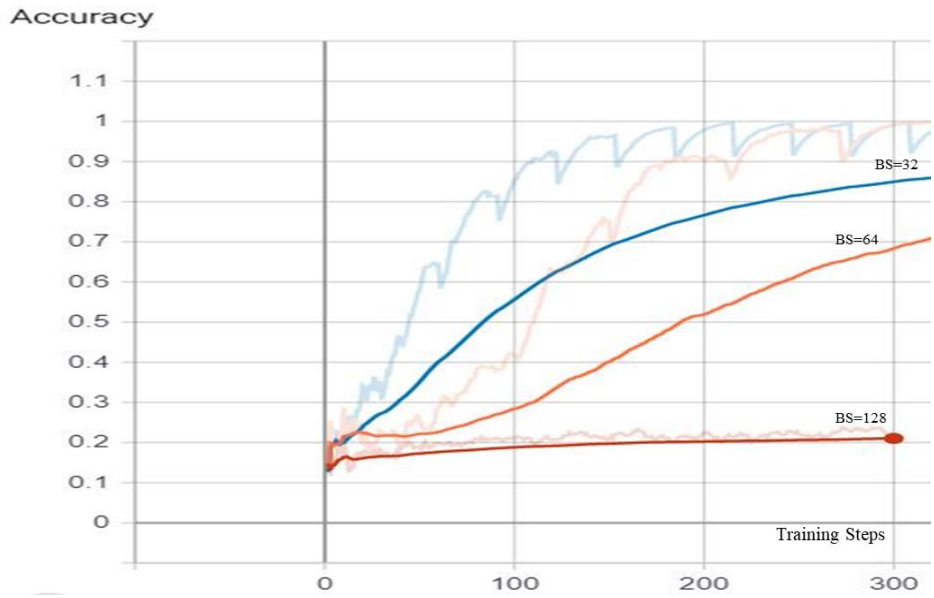
(a)



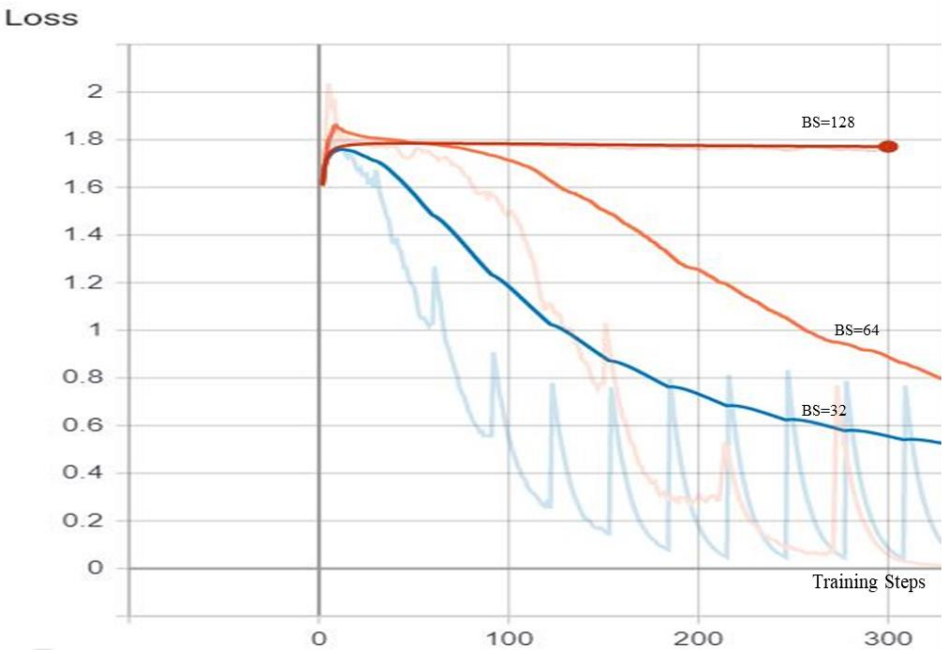
(b)

Figure 15 Accuracy (a) and Loss (b) plot on different learning rate

b. Batch Size: An aforementioned network model i.e. ensemble of AlexNet, GoogleNet and RetinaNet was trained through the ultrasonography Image of training dataset varying by batch size as 32, 64 and 128. The accuracy and loss value at every training step was recorded in a log file and then all values plotted as shown in figure16. As shown in figure16, training with 32 and 64 batch size converged and result high training and validation accuracy whereas with batch size with 128 results high training and validation loss and did not converge.



(a)

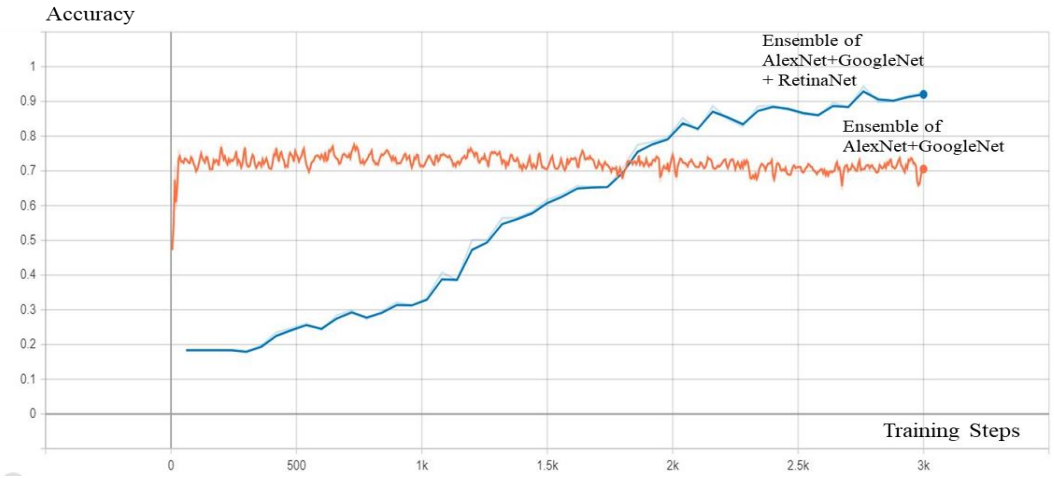


(b)

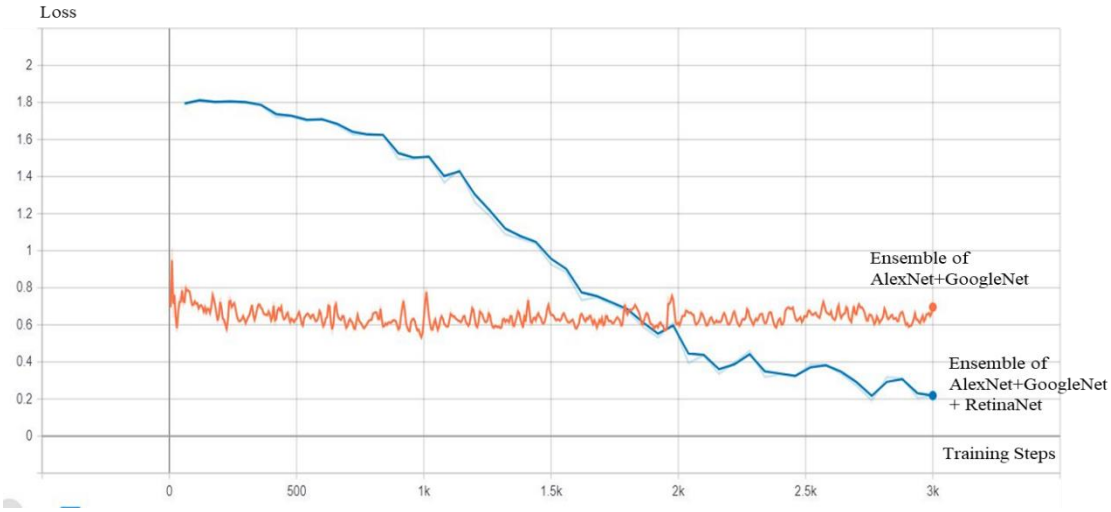
Figure 16 Accuracy(a) and Loss(b) plot on varying the Batch Size

5.3 Comparison with different Fine-tuned Ensemble Network Model

The ensemble AlexNet and GoogleNet CNN model was compared with the CNN model used in this research i.e. ensemble AlexNet and GoogleNet with RetinaNet. A similar experimental set up was done for both model such as hyperparameter learning rate set to 0.001, batch size to 32. The both ensemble models were trained through the same thyroid ultrasonography dataset and the training accuracy and loss values were recorded. The accuracy and loss graph of two different ensemble model is shown in figure 17.



(a)



(b)

Figure 17 Accuracy (a) and Loss(b) plot of different ensemble CNN model

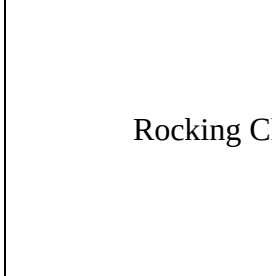
As shown in figure 17, the combination of AlexNet, GoogleNet and RetinaNet model had high accuracy and low training loss as compared to the combination of only AlexNet and GoogleNet. Hence the model used in this research had better performance for classifying ultrasonography image and ensemble RetinaNet framework can improve the performance of thyroid nodules recognition and classified the nodule more appropriately.

5.3 Thyroid Nodular Classification

By Using the preprocessed Ultrasonography images, the neural network was trained to classify the US image benign as TRAID2 and TRAID3 whereas malignant nodules as TRAID4a, TRAID4b, TRAID4c and TRAID5. The transfer learning method such AlexNet, GoogleNet and RetinaNet was used to train the model, which are the most popular image recognition model and has been previously successfully adapted for medical image analysis. The AlexNet, GoogleNet and RetinaNet model was pretrained with 1.2 million images labeled with 1000 semantic classes from the ImageNet Large Scale Visual Recognition Challenge repository. The AlexNet, GoogleNet and RetinaNet model architecture consists of the following layers which are pretrained, and contain information that can discriminate between images: a stem layer, Inception-A layers, Inception-B layers, Inception-B layer, Feature Pyramid Layer, a pooling layer, a dropout layer, a fully connected layer, and a softmax layer.

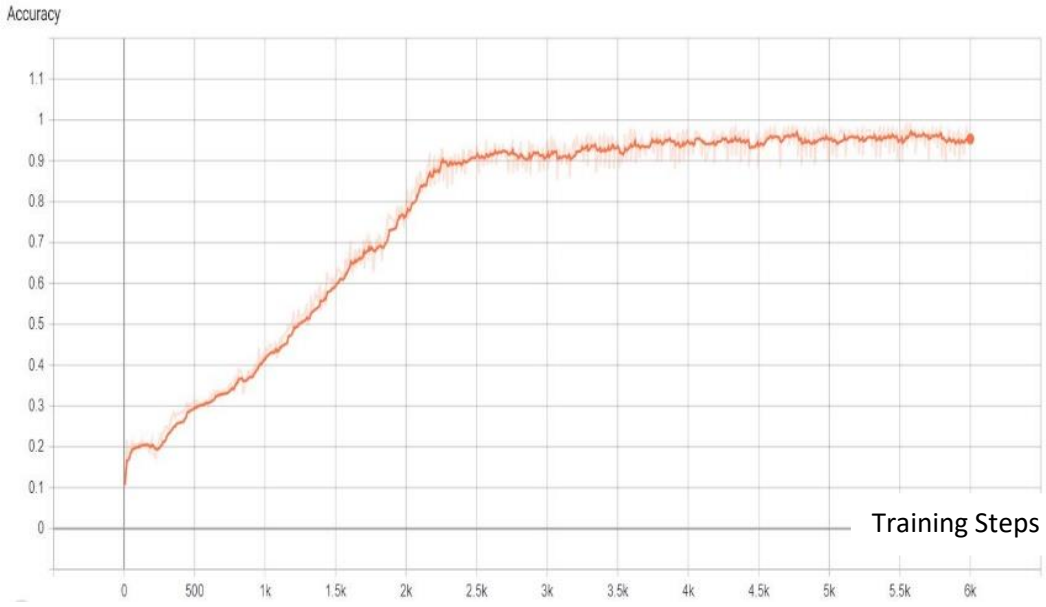
The pretrained AlexNet, GoogleNet and RetinaNet models were classified the natural Image successfully as shown in table 4:

Table 4: ImageNet dataset classification by Ensemble Network

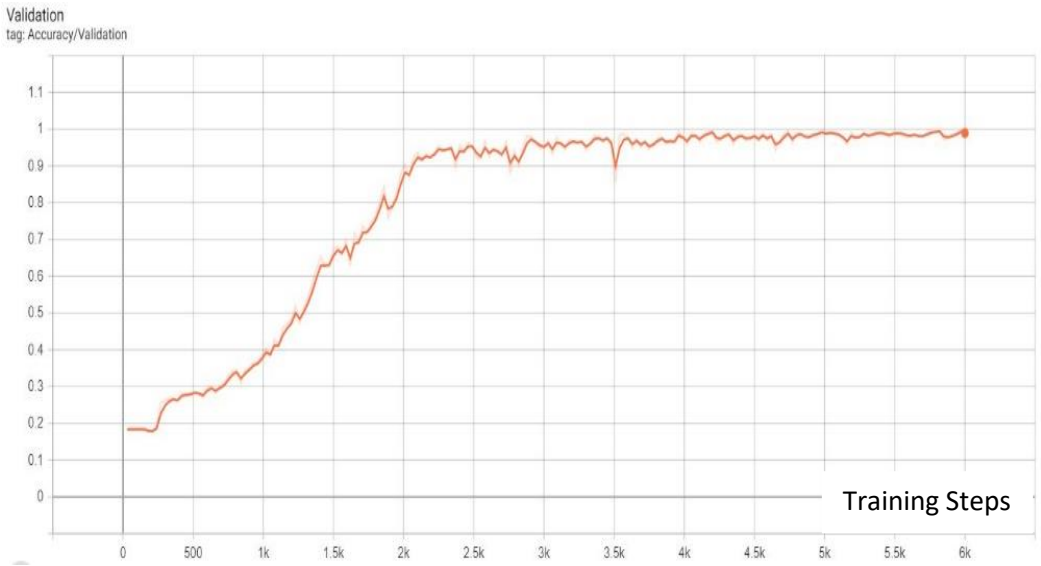
S.N.	Real Image of ImageNet Dataset	Predicted by the model
1.		Probability 94.929% - Rocking chair Probability 5.26% - Throne Probability 0.10% - Barber chair Probability 0.03% - Barber shop Probability 0.03% - Folding Chair
2.		Probability 30.21% - Magpie Probability 7.96% - Coucal

	Magpie	Probability 6.75% - Junco Probability 6.50% - Indigo Bounting Probability 3.37% - Jay

The dataset used for this study comes from open access database for thyroid nodule TDID (Thyroid Digital Image Database), which contains in total 480 valid cases and the images in the grayscale. Among the 480 cases with TIRADS score, 280 cases were diagnosed as malignant (TIRADS score 4a, 4b, 4c and 5) and 200 cases as Benign (TIRADS score 2 and 3). The image augmentation process such as rotation by 90 degree, Gaussian noise, flip and random colorization was used to produce 2000 number of datasets for training the convolutional Neural Network model. Among them 1400 images were used for training, 400 images used for validation and rest 200 images for test sets. The US Image was tuned on the Network model pretrained by ImageNet dataset to produce the fine-tuned fully connect network model to create a new fully connected layer. A "bottleneck layer," used with an extremely small number of units (compared with the adjacent layers). A small number of units can aggregate the propagated information and extract fundamental features from the input data. The new fully connected layer was trained with hyperparameters, with a learning rate of 0.001, a batch size of 32, model store frequency of 300, and 6000 training steps. The validation data was used in a holdout cross-validation manner.



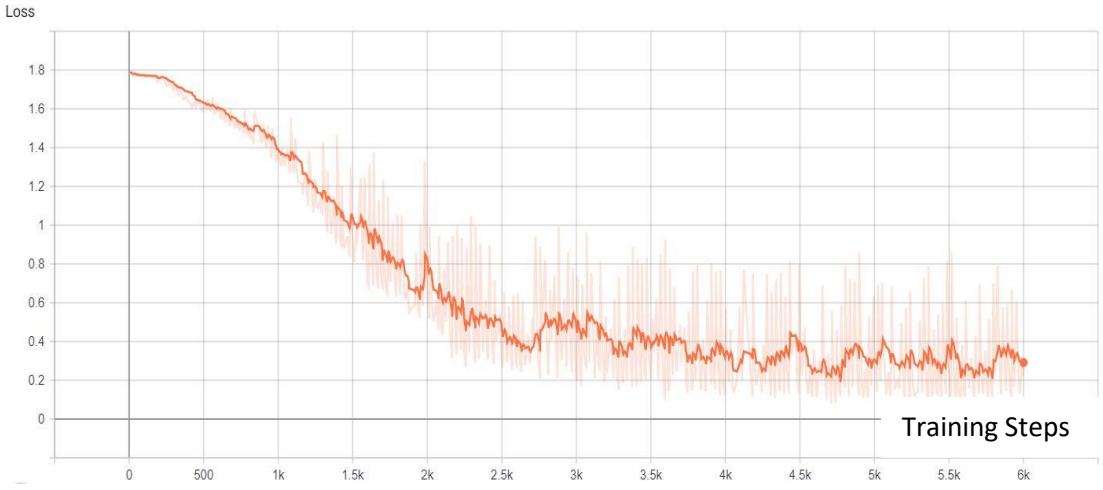
(a)



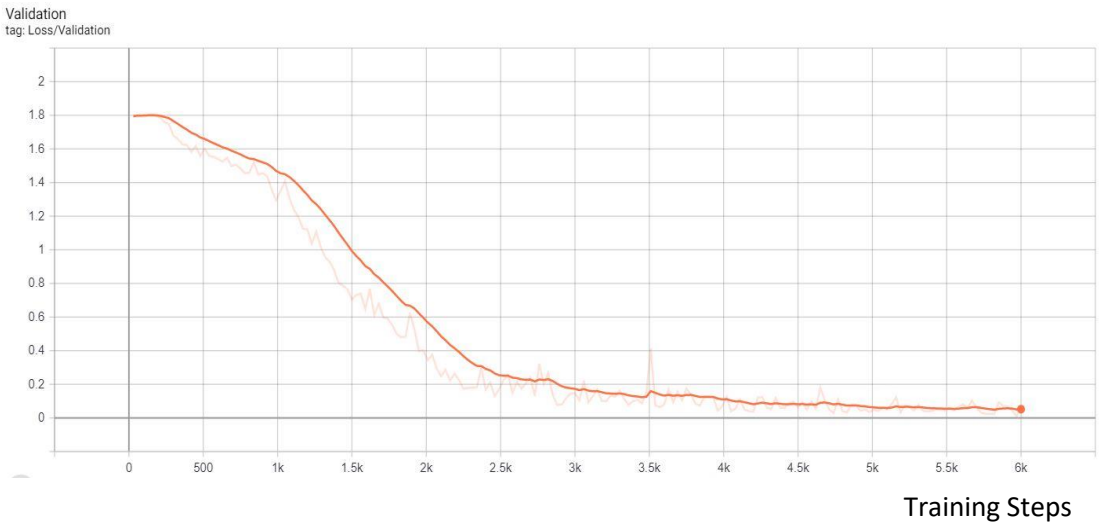
(b)

Figure 18: Training(a) and Validation(a) Accuracy of the given Network Model

The accuracy was recorded as training accuracy and validation accuracy in every 10 training steps for 6000 steps. The training step 2200 was identified as the point where the gap between the training accuracy and validation accuracy began to spread. So that training step 2200 was selected as the final model without overfitting. Benignity (TIRADS2, TIRADS3) or malignancy (TIRADS4a, TIRADS4b, TIRADS4c and TIRADS5) was presented based on a probability threshold of 0.85.



(a)



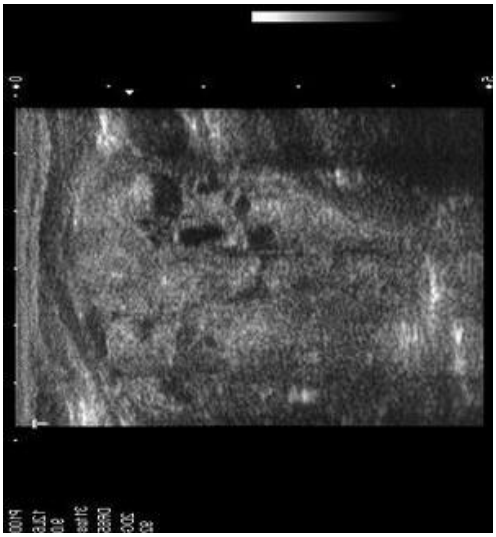
(b)

Figure 19: Training (a) and Validation (b) loss of the given model

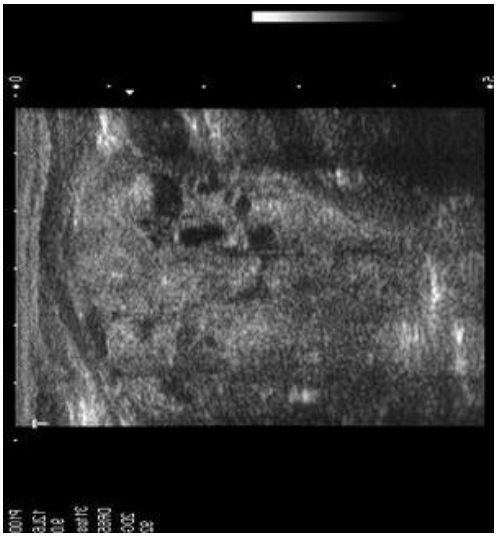
5.4 Test Result

The internal test datasets comprised 200 US nodule images having 64 images were benign and 136 images were malignant. Of the 64 benign nodule, 33 nodules were TIRADS2 and 31 nodules were TIRADS3 and of the 136 malignant nodules, 43 were TIRADS4a, 28 were TIRADS4b, 33 were TIRADS4c and 32 were TIRADS5.

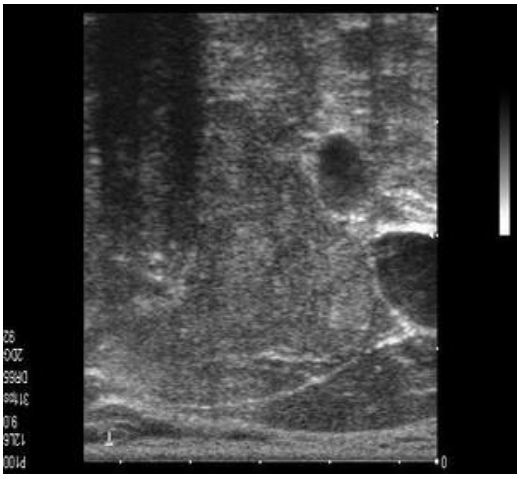
The test dataset run on the aforementioned fine-tuned ensemble CNN network and the outputs are observed. Some of the test result is shown in figure 20.



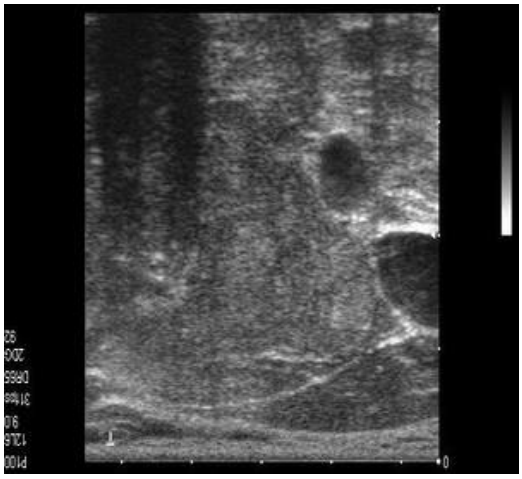
(a)Original Image of TIRADS 4a



(b) Model Predicted as TIARADS 4a



(c) Original Image of TIRADS 3



(d) Model Predicted as TIRADS 2

Figure 20 Prediction of ultrasonography of thyroid test image by the given model

The 200-test ultrasonography image of different thyroid nodule run on the fine-tuned ensemble AlexNet, GoogleNet and RetinaNet networks and the predicted result is shown in the form of Confusion matrix for multiple classification. Then Evaluation metric such as Precision, Recall and F1 score was calculated separately.

		Predicted						Precision	Recall	F1 Score
		TIRADS 2	TIRADS 3	TIRADS 4a	TIRADS 4b	TIRADS 4c	TIRADS 5			
Actual	TIRADS 2	31	1	0	0	1	0	0.9394	0.9117	0.9253
	TIRADS 3	1	28	0	1	1	0	0.9032	0.9032	0.9032
	TIRADS 4a	0	1	39	2	1	0	0.9069	0.9512	0.9285
	TIRADS 4b	1	1	0	25	0	1	0.8928	0.862	0.8771
	TIRADS 4c	0	0	1	0	31	1	0.9394	0.8857	0.9117
	TIRADS 5	0	0	1	1	1	29	0.9063	0.9354	0.9206

Figure 21 Confusion Matrix for Classification of Thyroid Nodules

The confusion matrix shown in figure 21 reflected that the model used in this research i.e. ensemble of AlexNet, GoogleNet and RetinaNet could not classify the thyroid ultrasonography image hundred percent but the overall performance score of the model is about 93% was satisfactory. The difference in classification done by senior radiologist of TDID dataset and the model used in this research was shown in table 5.

Table 5: Classification Performance of the given Fine-tuned Network

Thyroid Type	Classification by Radiologist (Based on TDID dataset)	Classification by Fine-tuned Network
TIRADS2	33	31
TIRADS3	31	28
TIRADS4a	43	39
TIRADS4b	28	25
TIRADS4c	33	31
TIRADS5	32	29

CHAPTER 6 CONCLUSION AND LIMITATION

6.1 Conclusion and Limitation

It has been explored the problem of thyroid nodule classification based on TIRADS. The overall process includes preprocessing of ultrasonography images, image augmentation and classification by ensemble of AlexNet and GoogleNet with one stage fast object detection technique such as RetinaNet. The performance parameters of the ensemble network model set up was calculated from the confusion matrix. The evaluation metrics such as precision which gives the proportion of predicted TIRADS was actually correct and value ranged from 89% to 94%. Another metric recall was calculated which gives the proportion of actual TIRADS was predicted correctly and its value ranged from 86% to 96%. Similarly, F1 score was calculated which gives the score that how much the mode is appropriate and its value ranged 87% to 93%. Hence the overall performance of the aforementioned ensembled network model was found satisfactory for the internal test dataset.

The performance of the aforementioned fine-tuned ensemble model is expected to increase by including more data and expanding the datasets to realistic data from the local hospital and diagnostic center. The features of different classes of thyroid nodule are less heterogenous to each other so that sometimes miss to predict correctly.

6.2 Future Works

In future research, it is planned to apply this method of image classification to various real and local images for in-depth analysis of Ultrasonography images to gain a better comprehension result. The benign and malignant thyroid nodules are very similar to each other and classify them using the factor such as nodule composition, echogenicity, shape and calcification. So identifying the region of interest (ROI) of the affected area which further improves the performance of the given model.

REFERENCES

[1]. Qing Li, Weidong Cai, Xiaogang Wang, Yun Zhou, David Dagan Feng and Mei Chen, "Medical Image Classification with Convolutional Neural Network," 13th International Conference on Control, Automation, Robotics & Vision, 2014.

[2]. Kaiming He Xiangyu Zhang Shaoqing Ren Jian Sun, "Deep Residual Learning for Image Recognition", IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[3]. Ashnil Kumar, Jinman Kim, David Lyndon, Michael Fulham, and Dagan Feng, "An Ensemble of Fine-Tuned Convolutional Neural Networks for Medical Image Classification", IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS, 2016.

[4]. Xueyan Mei, Xiaomeng Dong, Timothy Deyer, Jingyi Zeng, Theodore Trafails and Yang Fang, "Thyroid Nodule Benignity Prediction by Deep Feature Extraction", IEEE 17th International Conference on Bioinformatic and Bioengineering, 2017.

[5]. Ye Zhu, Zhuang Fu and Jian Fei, " An Image Augmentation Method Using Convolution Network for Thyroid Nodule Classification by Transfer Learning", 3rd IEEE International Conference on Computer and Communication, 2017.

[6]. Wenfeng Song, Shuai Li, Ji Liu, Hong Qin, Bo Zhang, Shuyang Zhang, and Aimin Hao, "Multi-task Cascade Convolution Neural Networks for Automatic Thyroid Nodule Detection and Recognition", IEEE Journal of Biomedical and Health Informatics, 2015.

[7]. Jingfeng Lu, Wanyu Liu Department of Automatic Measurement and Control Harbin Institute of Technology Harbin, China, "Unsupervised Super-Resolution Framework for Medical Ultrasound Images Using Dilated Convolutional Neural Networks", 3rd IEEE International Conference on Image, Vision and Computing, 2018.

[8]. Hweijin Jungid, Bumsoo Kim, Inyeop Lee, Minhwan Yoo, Junhyun Lee, Soooyound Hamid, Okhee Woo and Jaewoo Kang, "Detection of Masses in

Mammograms using a One-Stage Object Detector based on a deep convolution Neural Network", Research article on PLOS one, 2018

[9]. Nima Tajbakhsh_, Member, IEEE, Jae Y. Shin_, Suryakanth R. Gurudu, R. Todd Hurst, Christopher B. Kendall, Michael B. Gotway, and Jianming Liang," Convolutional Neural Networks for Medical Image Analysis: Fine Tuning or Full Training?" IEEE Transactions on Medical Imaging, 2016.

[10]. Kaiming He Xiangyu Zhang Shaoqing Ren Jian Sun, "Deep Residual Learning for Image Recognition", IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[11]. Rodrigo G. F. Soares, Emeson J. S. Pereira, "On the performance of pairings of activation and loss functions in neural networks" International Joint Conference on Neural Networks (IJCNN), IEEE, 2016.