



A Novel Lightweight Deep Learning-Based Histopathological Image Classification Model for IoMT

Koyel Datta Gupta¹ · Deepak Kumar Sharma²  · Shakib Ahmed² · Harsh Gupta² · Deepak Gupta³ · Ching-Hsien Hsu^{4,5,6}

Accepted: 2 June 2021

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract

The unavailability of appropriate mechanisms for timely detection of diseases and successive treatment causes the death of a large number of people around the globe. The timely diagnosis of grave diseases like different forms of cancer and other life-threatening diseases can save a valuable life or at least extend the life span of an afflicted individual. The advancement of the Internet of Medical Things (IoMT) enabled healthcare technologies can provide effective medical facilities to the population and contribute greatly towards the recuperation of patients. The usage of IoMT in the diagnosis and study of histopathological images can enable real-time identification of diseases and corresponding remedial actions can be taken to save an affected individual. This can be achieved by the use of imaging apparatus with the capacity of auto-analysis of captured images. However, most deep learning-based image classifying models are bulk in size and are inappropriate for use in IoT based imaging devices. The objective of this research work is to design a deep learning-based lightweight model suitable for histopathological image analysis with appreciable accuracy. This paper presents a novel lightweight deep learning-based model "Reduced-FireNet", for auto-classification of histopathological images. The proposed method attained a mean accuracy of 96.88% and an F1 score of 0.968 on evaluating an actual histopathological image data set. The results are encouraging, considering the complexity of histopathological images. In addition to the high accuracy the lightweight design (size in few KBs) of the ReducedFireNet model, makes it suitable for IoMT imaging equipment. The simulation results show the proposed model has computational requirement of 0.201 GFLOPS and has a mere size of only 0.391 MB.

Keywords Deep learning · Disease diagnosis · Histopathological · Image classification · IoMT

✉ Deepak Kumar Sharma
dk.sharma1982@yahoo.com

Extended author information available on the last page of the article

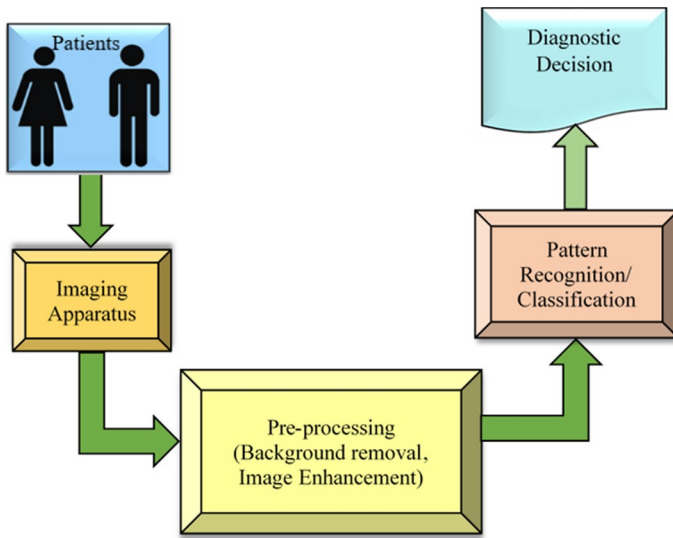


Fig. 1 Computer-assisted disease diagnosis using medical images

1 Introduction

Improper medical examination, failure to turn up for the follow-up, difficulty to access the medical records of an individual can lead to delay in detection of diseases [1]. The IoT technology that has been applied to an extensive variety of services is also being applied to the health sector. IoT is certainly redefining the healthcare service by transforming the medical devices and people interface in providing medical services. IoT based healthcare applications are beneficial for patients, medical professionals, hospitals, pharmaceutical, and insurance companies. The IoT based healthcare applications are highly important since it improves medical care provided to the patients and reduces the cost of such facility [2]. The IoMT allows effective and efficient supervision of patient health and can even make an early diagnosis of disease and save an individual's life with the initiation of timely treatment [3]. The availability of new-age sensors has improved the performances of IoMT services by enabling the timely and precise collection of an individual's physiological parameters. However, the accuracy of the diagnosis system is not only dependent on precise data but also on the image analysis technique. Researchers [4] have also developed deep learning-based frameworks for task scheduling and sequencing for IoT assisted medical systems.

In the orthodox method of disease diagnosis, the procedures involved have huge costs and are time-consuming. The pathological sample has to be collected by a trained individual and studied by pathologists. The pathologist's prepared report is based on the observation of the samples and the reports are referred by practitioners. Lack of enough number of trained pathologists at times can delay the process of proper diagnosis of diseases and can hinder appropriate medical intervention at the right time endangering the patient's life. This raises the requirement of auto-detection of medical images. The usage of computer-assisted detection of diseases can be tracked down to the 1990s [5]. The use of artificial intelligence in the analysis of medical images for the medical evaluation of an individual proved to be immensely useful (Fig. 1). With the enormous growth in the field of artificial intelligence and machine learning, the analysis of

pathological images has become one prime area of focus. The need for efficient computer-assisted disease diagnosis is on the rise with the rise in critical disease cases all over the world. Automated analysis of digital images of a tissue sample is quite challenging since the cellular structure in images varies in terms of colour, shapes, sizes, and other physiological characteristics. The use of deep learning in computer-assisted diagnosis has improved the precision of the screening of diseases. With the advent of highly accurate learning networks like Convolutional Neural Network (ConvNet/CNN) [6] and recurrent neural network (RNN) [7], it is possible to detect diseases using histological tissue images.

One of the major emphases of examination of pathological images has been on the computerized analysis of cytological images for disease detection. These images are mostly characterized by lone or clustered cells and are easier to process as compared to histopathological samples. The histopathological images provide a more extensive picture of the disease and its consequences on tissue samples. Moreover, screening of histopathological images for disease diagnosis is considered to be the gold standard, since it is capable of detecting a large number of diseases and several types of cancers [8]. However, the additional pathological traits present in these images pose new complexities in the process of automated disease screening. These complexities can be resolved by using sophisticated AI-based approaches. These image analyzing methods can assist the medical practitioners in making an exact diagnosis of the ailment and categorize morphological characteristics correlated with the prognosis. In IoMT applications, such automated image analysis can play a vital role in the early and accurate detection of diseases, but due to the immense size of such software, the computation is possible only either on fog nodes or on cloud servers. To compute the images locally, lightweight software is needed which can be embedded in IoMT devices. Many of the existing works as specified in Sect. 2, have achieved high accuracy in disease detection but are computationally expensive making them unsuitable for deploying within medical devices. This paper proposes a novel deep learning model ReducedFireNet, with substantially reduced size and a considerably low computational requirement for analyzing histopathological samples. Majority voting [9] is deployed to generate the final prognosis. The model is further compressed using quantization [10] without any substantial degradation in the performance of the proposed model. The key contribution of this research work can be summarized as follows:

- A deep learning-based histopathological image processing model is proposed that learns important traits from real-life samples essential for the diagnosis of diseases.
- A novel model is proposed with a size of a few KBs and a requirement of low computational power, such that it can be effectively embedded into any medical image capturing device and can be used to process the information at the source.
- The simulation results validate the findings of the proposed “ReducedFireNet” model when applied to a real-life medical dataset. The comparative statement specified in the result section proves the efficiency of the model in disease prognosis using histopathological images.

The remaining sections of this paper are arranged as follows: Sect. 2 explains the related works on the proposed problem; the subsequent section discusses the motivation behind the work. Section 4 presents the proposed solution. The experimentation and the result analysis are described in Sect. 5 and Sect. 6 respectively, and, in the end, the paper is concluded in Sect. 7.

2 Related Work

There are diverse techniques of medical imaging procedures that include ultrasonography (USG), computed-tomography (CT), magnetic-resonance imaging (MRI), and digitally scanned histology images for studying medical cases. The last few decades have witnessed incredible growth in the area of medical image analysis using the deep learning approach. To prevent the deaths of patients due to late diagnosis, the research community has been dedicated to modelling AI-based frameworks for the diagnosis of fatal diseases specifically different forms of cancer.

Sun et al. [11] presented an adaptive fuzzy C-means-based mass detection approach along with a supervised neural network to inspect the presence of tumors in an area. The study aims to show the importance of the ipsilateral multi-view CAD mechanism along with concurrent analysis to reduce false-positive rates. To improve the competence of a breast-cancer CAD system, Kumar et al. [12] offered Zernike moments (ZMs) image retrieval system. To perform segmentation of breast tumors, Saidin et al. [13] used pixels and a region growing method. A scheduling framework was proposed for the grid resource allocation. Xu et al. [14] performed a coarse segmentation followed by recognition of image edges. The edge detection was performed using the mean gray-scale value as a means for combining regions. A distance-based computation is done for internal markers and morphological dilation was applied for the external marker. In recent times authors in [15] presented an extreme learning machine (ELM) model for the prognosis of breast cancer. In addition, a gain ratio feature selection method is deployed to remove insignificant features. A cloud computing-based system enabled with ELM is also developed for remote diagnosis of breast cancer. The research work reported accuracy of 98.68% using the Wisconsin Diagnostic Breast Cancer dataset.

Huang et al. [16] presented a Machine learning aided Ultrasound CAD for recognition of fetal standard plane. In [17] Doyle et al. presented a Graph Embedding algorithm to distribute the various grades of prostate cancer. The researchers deployed an SVM classifier which was able to achieve the highest accuracy of 92.8% in differentiating between the various cancer grades. Kim et al. [18] presented a CNN and U-Net-based automatic estimation method for measuring the abdominal circumference of a fetus. In [19], the authors managed to achieve an accuracy of 97% for grading prostate cancer. They used H&E (hematoxylin and eosin) stained samples for deriving features of nuclear structures. Rajpoot et al. [20] present manifold learning for shape-based differentiation of “prostate nuclei”. The researchers presented a CAD method for the detection of prostate cancer from high-quality diverse MRI images [21]. An accuracy of 62.3-76.5% is reported in the classifying of H&E-stained cervical tissue in [22]. Roy et al. [23] presented a lung cancer detection model with an accuracy of 94.12%. The proposed model used a fuzzy inference system for the prognosis and gray-scale transformation for enhancing contrast.

In [24], the authors proposed a naive Bayes network-based classification model for lymphoma. A two-stage framework is presented, the first stage transforms the raw pixel-level information to spectral planes and then global features were calculated on each spectral plane. Classification of lesions with stationery wavelet transformation-based descriptors are employed in [25]. The Analysis of Variance reported an accuracy of 100%. Nascimento et al. [26] presented a CAD prototype based on non-morphological and morphological features to diagnose subtypes of lymphoma. The model exhibited an accuracy between 94 and 96%. K. H. Abdulkareem et al. [27] proposed a machine learning and IoT based framework to diagnose COVID-19. The study uses

Naïve Bayes, Support Vector Machine and random forest for the classification task and reports the highest achieved accuracy (SVM) to be 95%.

Al-Waisy et al. [28] developed a deep learning-based diagnosis tool for detecting COVID-19 based on X-ray images of an individual's chest. The proposed deep net system reported accuracy of 99.93%. Researchers in [29] uses ResNet-14 architecture of CNN to detect anterior cruciate ligament injury from an MRI (Magnetic resonance imaging) and reported an average accuracy of 92%.

The research works discussed in this section are mostly based on complex frameworks which have high computational requirement. In the case of simple models like SVM or Naïve Bayes, deployed for disease prognosis, the accuracy is relatively lower than deep learning-based models.

3 Motivation

The global community is witnessing a very difficult situation due to COVID-19. This has further raised the importance of the Internet of Medical Things in providing essential healthcare services to patients, remotely. People who are suffering from life-threatening diseases like cancer, chronic kidney disorder, cirrhosis, and alike can have a better life expectancy if proper timely medical care is available. Diverse forms of medical image analysis are available for images available through USG, CT or MRI scan, virtual microscopy, whole slide scanning techniques. There has been an increased interest in utilizing machine learning and deep learning for medical applications however very few approaches are being developed to take advantage of the IoMT devices for improving medical pipelines. Most of the research work utilizing deep learning and statistical methods used for medical diagnosis is not suitable for being embedded in the medical imaging devices due to their large size and high computational requirement which lead to IoMT devices being ignored by many researchers.

The motivation of the proposed work is to develop a lightweight histopathological image classification system using convolutional neural networks that could utilize the weak computational power and low storage capacity of IoMT devices that are slowly being integrated into medical equipment. The importance of the availability of a local automated image analyzer is immense since it ensures early diagnosis of disease without transferring the data to the next level for computation. The automated analyzer provides an early estimation of the disease and enables the medical practitioner to provide timely appropriate care to the patients. Apart from using IoMT devices to help lab technicians during the diagnosis of a disease, another objective of the proposed work is to develop a classification process in such a manner that will protect the patient's medical data since the process of classification takes place on the IoMT device itself, there is no need to send the medical data to an external server. The proposed work's success will allow us to use similar techniques on other medical classification applications, we could run multiple low-cost systems of our proposed model for different diseases at the same time on the patient's medical data, and this will greatly increase the chance of detecting an ailment that would have gone unnoticed earlier.

4 Proposed Solution

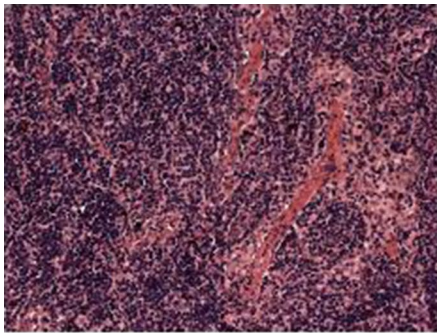
The importance of local automated analysis of histopathological images in IoMT applications has been emphasized in the earlier sections. Such an automated image analyzer model requires a solution with very low memory and computational needs while having a minimal drop in accuracy when compared to the existing state of the art models for image classification. Our proposed disease prognosis model is a lightweight image-based classifier with minimal resource requirement and high accuracy. The proposed solution is divided into 4 steps. Firstly, data augmentation is performed to improve the available dataset and the images are broken down into smaller patches. These patches are used to train our proposed model called ReducedFireNet. The concept of majority voting is employed to produce the final prediction to the corresponding image. The model is further compressed to a smaller size using Quantization.

In this section, we discuss each of these approaches in the context of the proposed solution.

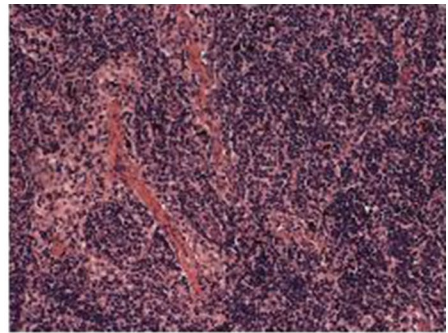
4.1 Data Augmentation and Patch Creation

Data augmentation [30] is the technique used to make the dataset more diverse and expand the size of the dataset artificially by applying meaningful transformations to the existing data. It also helps in solving the class imbalance problem that may exist in datasets. In image classification tasks, having a diverse and balanced dataset is very important as datasets are the lifeline of our systems, biased or imbalanced datasets can lead to fatal errors in our predictions. Some of the data augmentation methods which can be applied to images are rotation, shearing, brightness shift, random zoom, horizontal flip, vertical flip, etc. It is however important to note that not all data augmentation techniques can be applied to all types of data. We should pick the data augmentation methods in such a manner that the transformations should result in realistic images such that the label of the original image remains preserved. After carefully analyzing various data augmentations methods, we have decided to use horizontal flip, vertical flip, and brightness shift augmentations because these transformations will result in images that are most likely to be present in an unknown input image. The examples of these augmentations are shown in Fig. 2. Figure 2 shows the different augmentations performed to generate the final image dataset. To augment the image, the image can be rotated by 180° along the x-axis (Fig. 2b) or the y-axis (Fig. 2c) or a random brightness shift may be applied (Fig. 2d). There are other networks like Generative Adversarial Networks (GANs) [31] which can be utilized to create synthetic data samples that can substitute real data.

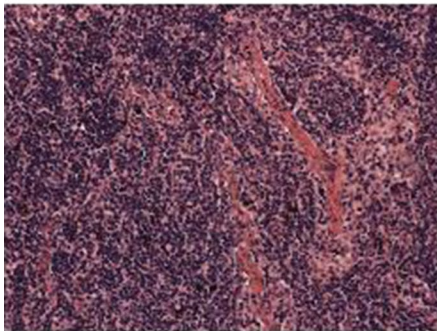
After applying data augmentation techniques, the image is broken down into patches. The entire process of data augmentation and patch creation is explained in Algorithm 1.



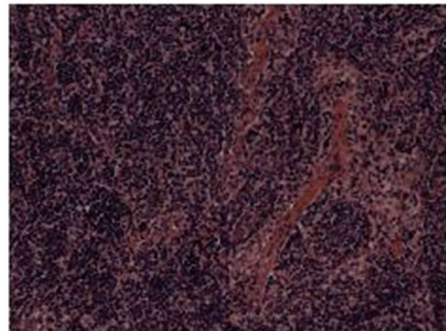
(a) Original Image



(b) Horizontal Flip



(c) Vertical Flip



(d) Brightness Shift

Fig. 2 Different data augmentation techniques for the same image

Algorithm 1: Data Augmentation & Patch Creation**Input:** Original Dataset $D_{original}$ $D_{synthetic} \leftarrow \{\}$ **For each class in $D_{original}$ do** $D_{class} \leftarrow D_{original} [class]$ Create new synthetic images from D_{class} using image augments $D_{class_synthetic} \leftarrow D_{class} \cup (\text{Augmented synthetic images})$ $D_{synthetic} \leftarrow D_{synthetic} \cup D_{class_synthetic}$ **End**Split $D_{synthetic}$ into D_{train} , D_{test} $D_{train_patches} \leftarrow \{\}$ **For each x_i, y_i in D_{train} do**Create patches of x_i $D_{train_patches} = D_{train_patches} \cup \{\text{patches}, y_i\}$ **End****4.2 Proposed Model—ReducedFireNet**

Our proposed model, ReducedFireNet is a type of Convolutional Neural Network (CNN), CNN is the backbone of all state-of-the-art models used for image classification. They are a variation of neural networks where matrix multiplication is replaced by convolution for at least one layer of the model. Unlike a neural network which uses each pixel of the input image as an independent input, convolution allows the neighbourhood pixels to be taken into consideration which drastically improves the network's performance. The Reduced-FireNet model uses multiple Fire modules., which were originally the building blocks of SqueezeNet [32]. SqueezeNet is a CNN model developed by Iandola et al. in an attempt to reduce the model size while maintaining accuracy when compared to AlexNet [33]. The SqueezeNet model mainly consists of 1×1 and 3×3 filters. They efficiently used 1×1 filters to reduce the input channels to 3×3 filters and used the strategy of late downsampling in the network. Using 1×1 filters allows us to reduce the number of channels. If we have an input of size $32 \times 32 \times 4$ and is passed to a convolutional layer containing 2 filters each of size $1 \times 1 \times 64$, then the output will be of size $32 \times 32 \times 2$, i.e. the number of channels is reduced from 4 to 2 as shown in Fig. 3.

Using 1×1 filters and decreasing the input channels to 3×3 filters help in reducing the number of parameters and downsampling late in the network helps in achieving higher accuracy.

The fire module consists of 2 main layers, the first being a 1×1 convolutional layer and the second being a concatenation of 1×1 and 3×3 convolutional layers (Fig. 4). The number of filters in each of these layers can be set according to the need with the only

Fig. 3 Reduction of the number of channels through 1×1 convolution

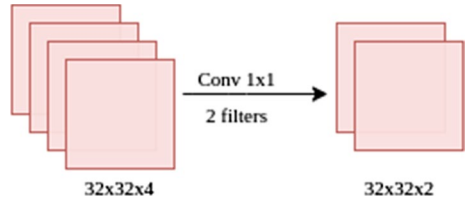
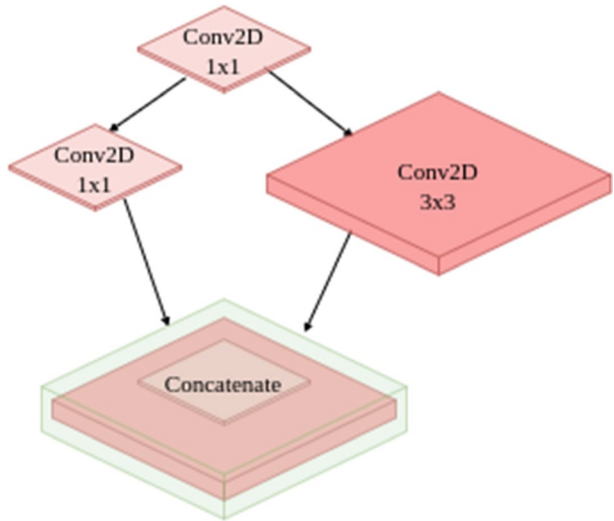


Fig. 4 A single Fire Module



restriction being that the number of 1×1 filters in the first layer should be less than the sum of the number of 1×1 and 3×3 filters of the second layer.

Our proposed ReducedFireNet model consists of 4 Fire modules. The entire model creation process is explained in Algorithm 2.

Algorithm 2: Model Initialization

Initialize_FireModule():

Create: First Conv2D Layer
Create: Second Conv2D Layer
Create: Third Conv2D Layer
Create: Concatenate (second conv2D, third conv2D)

Initialize_ReducedFireNet():

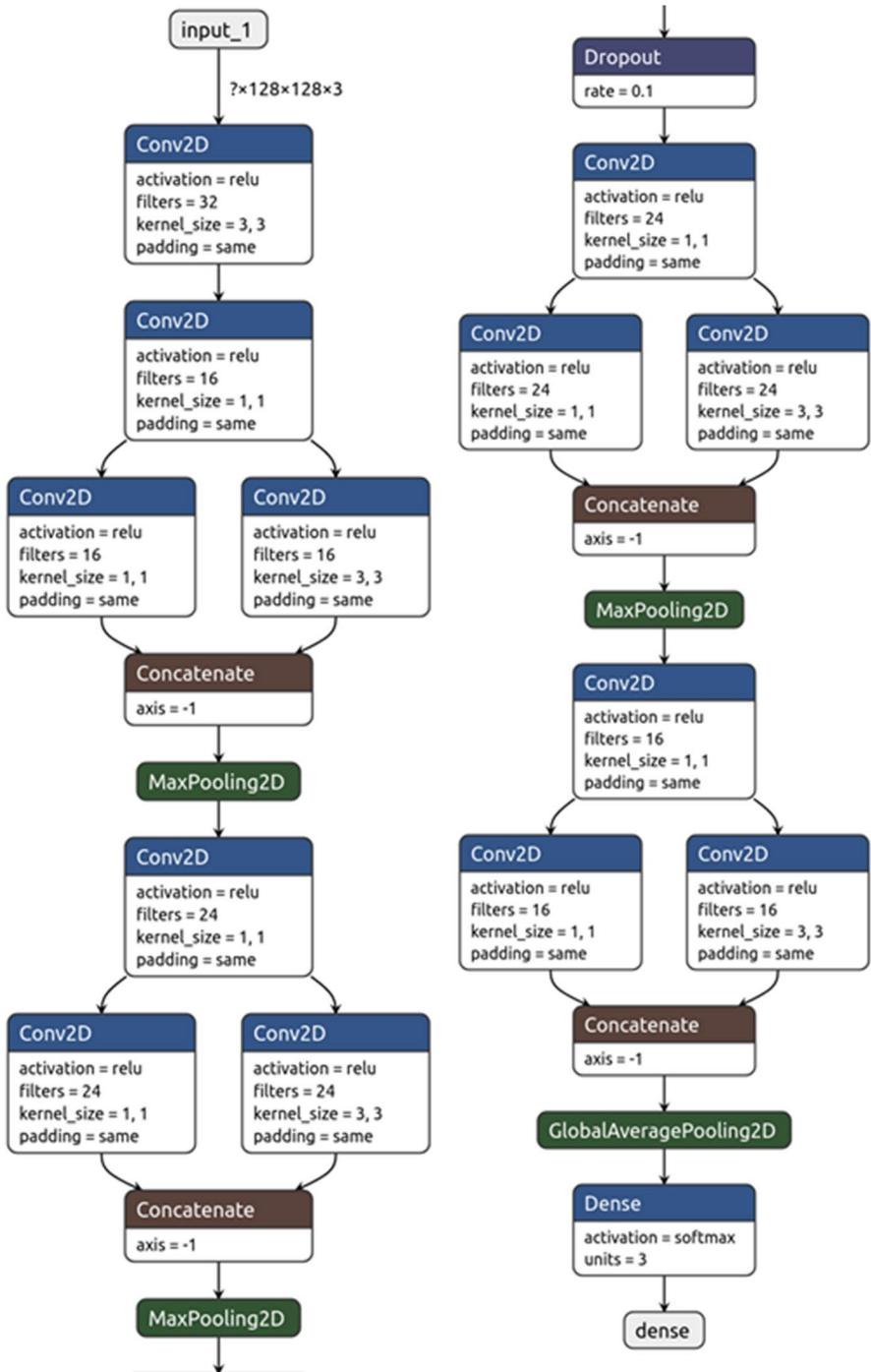
Create: Input Layer
Create: Conv2D Layer
Create: First Fire Module
Create: MaxPooling2D Layer
Create: Second Fire Module
Create: MaxPooling2D Layer
Create: Dropout
Create: Third Fire Module
Create: MaxPooling2D Layer
Create: Fourth Fire Module
Create: GlobalAveragePooling2D Layer
Create: Output Layer

Max-pooling layers are applied after every Fire module. They reduce the size of the input, speed up the computation and help in detecting more robust features. A dropout [34] layer is applied after the second fire module, to reduce overfitting and make the model more robust to the new inputs. Rectified Linear Unit (ReLU) [33] activation function is applied to the output of each convolutional layer. It gives an output 0 if the input is negative, else it directly outputs the input. The final Dense layer which is the output layer consists of 3 units corresponding to each of the output class and the activation function which is applied is Softmax[33]. The softmax function (Eq. 1) is a mathematical function that converts a numeric vector to a vector of real values which are interpreted as a probability, it applies an exponential function to each element of the input vector which is divided by the sum of all the exponents producing a normalized output where each output element lies between 0 and 1. It allows the output of penultimate layers, a real-valued vector to be represented as a normalized probability distribution. Softmax function is defined by the formula:

$$\text{softmax}(x)_a = \frac{\exp(x_a)}{\sum_a \exp(x_a)} \quad (1)$$

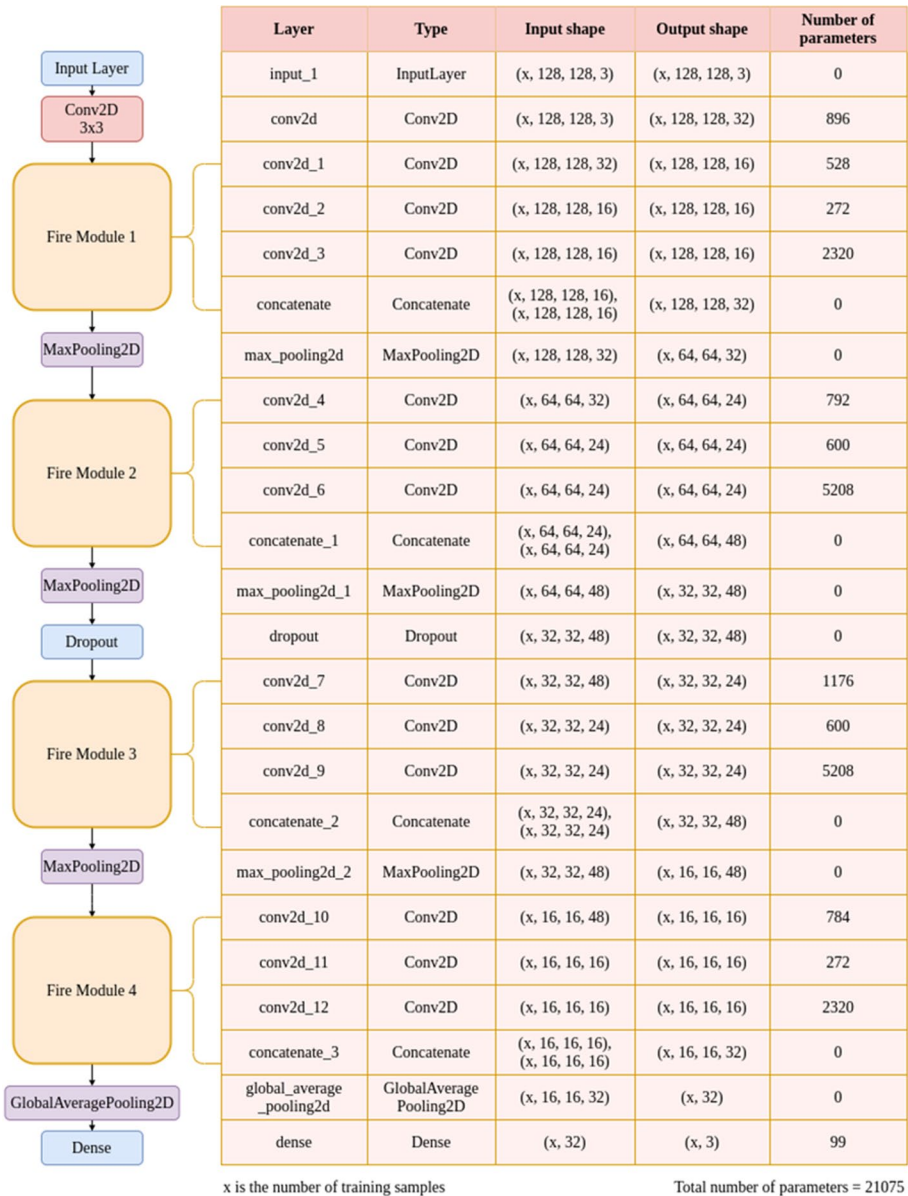
The comprehensive architecture of ReducedFireNet is shown in Fig. 5. The model is visualized using Netron [35]. In Fig. 5(a), ‘?’ in the output of “input_1” module represents the number of training samples.

Our proposed CNN model (ReducedFireNet) uses four Fire modules. A single Fire module consists of 2 layers, the first one is a 1×1 convolutional layer and the second one



(a) ReducedFireNet configuration

Fig. 5 a ReducedFireNet configuration b: ReducedFireNet architectural dimensions



(b) ReducedFireNet architectural dimensions

Fig. 5 (continued)

is the concatenation of 1×1 and 3×3 convolutional layers. These convolutional layers use ReLU activation functions. Each of the first three Fire modules is followed by a MaxPooling layer and the last Fire module is followed by a GlobalAveragePooling layer. A dropout layer is applied after the second MaxPooling layer. Finally, a dense layer with softmax as an activation function is applied for classification purpose.

4.3 Majority Voting

The concept of Majority Voting is borrowed from the ensemble-based classifiers. Ensemble-based classifiers utilize multiple base models to generate an optimal optimized model that performs exceptionally better than the individual base models. Simply in Majority Voting, predictions are performed for each patch of the image and the final prediction is the outcome with the largest count among the patch predictions [9]. As shown in Fig. 6, a high-resolution medical image has been divided into multiple image patches. Our model provides a prediction for every patch and the prediction with the most count is considered as the final prediction for the given image. The entire process of model evaluation using majority voting is explained in Algorithm 3.

Algorithm 3: Model evaluation using Majority Voting

Input: Train dataset $D_{train_patches}$, Test dataset D_{test}

$ReducedFireNet \leftarrow Initialize_ReducedFireNet ()$
 Train $ReducedFireNet$ using $D_{train_patches}$

$y_{prediction} \leftarrow \{ \}$

for each x_i **in** D_{test} **do**

$patches \leftarrow \{Break\ x_i\ into\ smaller\ patch\ images\}$

$y_{curr_prediction} \leftarrow \{ \}$

for each $patch$ **in** $patches$ **do**

$y_{curr_prediction} \leftarrow y_{curr_prediction} \cup \{predicted\ class\ of\ patch\}$

$y_{prediction} \leftarrow y_{prediction} \cup \{majority\ voting\ on\ y_{curr_prediction}\}$

End

$y_{true} = \{for\ each\ y_i\ in\ D_{test}\}$

Calculate accuracy, F1Score using $y_{true}, y_{prediction}$

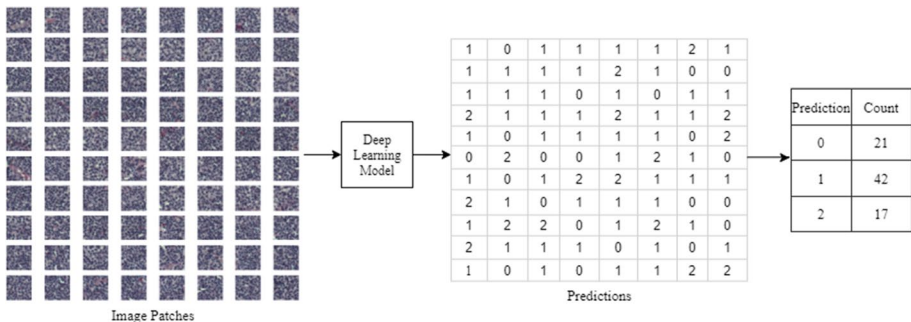


Fig. 6 Example of majority voting

4.4 Compression

Mobile and IoT Devices have limited computational power and memory space, hence there is a need to further compress our model. There are two main approaches used to compress a machine learning model, Pruning and Quantization [10]. Pruning efficiently reduces the size of a neural network by removing redundant connections that are present in a neural network, whereas Quantization is the process of transforming a machine learning model's parameter such that the model can be trained and executed at a lower approximated precision operation. Quantization decreases the size of the model by decreasing the number of bits required to represent the weights of the model. To limit the number of effective weights, many weights are shared by different connections and the weights are fine-tuned further to maintain high accuracy. Affine representation converts higher precision weights to lower precision values as shown in Fig. 7. The other benefits of having lower precision operations apart from a reduction in model size are faster execution, reduced power consumption, and reduced hardware cost as it is cheaper to create circuitry for lower precision data when compared to the original higher precision data.

The process of quantization is performed in 3 steps:

Step 1: A transfer function that converts data from a higher precision to a lower precision.

Step 2: The conversion process from the original model to the new compressed model is performed.

Step 3: Calibration is performed to compute new data required by the compressed model and fine-tune parameters if required by the model.

5 Experimentation

This section describes the implementation of our proposed solution. It is divided into 4 subsections. In the first subsection, we introduce the dataset which we will be using throughout our experiments along with various complications with medical datasets and how we have tried to solve them. Then we explain the training and evaluation of the ReducedFireNet model. In the third subsection, we analyze and compare the results of our model with various state of the art CNN architectures. In the last subsection, compression of the model is performed to reduce the memory and computational requirements. TensorFlow [36] and Keras [37] were used in our experiments.

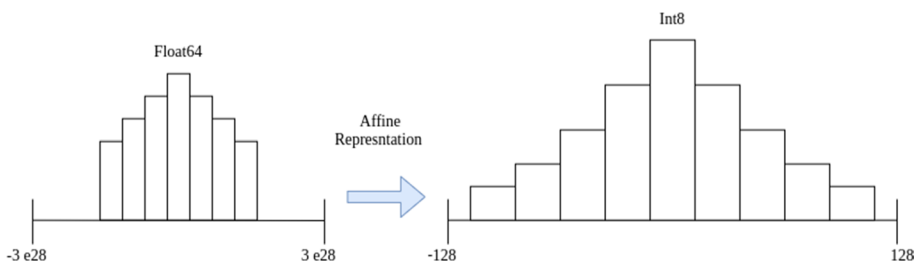


Fig. 7 Affine representation

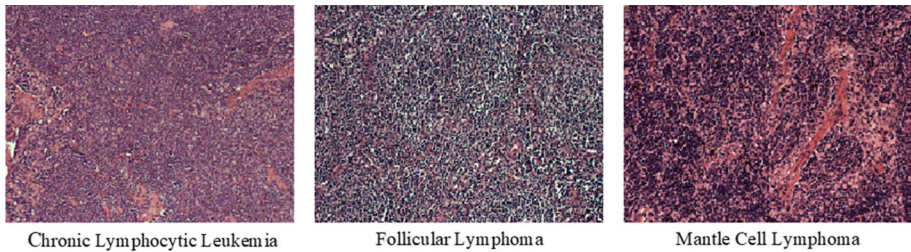


Fig. 8 An example image of each type of lymphoma

Table 1 Image distribution before and after augmentation

Lymphoma subtypes	Before augmentation	After augmentation
CLL	113	150
FL	139	150
MCL	122	150

5.1 Dataset, Augmentation, and Patch Creation

The Malignant Lymphoma dataset [38] which we have used consists of 113 images of chronic lymphocytic leukemia (CLL), 122 images of mantle cell lymphoma (MCL), and 139 images of follicular lymphoma (FL), in total 374 histopathological images stained with H&E. A sample image of each subtype is shown in Fig. 8.

Data used to train state of the art image classifiers are incredibly different from the available medical data. There is a severe lack of labelled medical datasets. For example, ImageNet [39] Dataset has over 14 million images with over 20,000 different subcategories. However, there is a lack of similar medical datasets available to the general public. Moreover, the creation of a similar combined medical dataset is challenging as most of the medical diseases are different from each other and may require an entirely different procedure for their identification. Further these available medical images usually have very high resolution, and the general practice of image resizing to a lower size is invalid for them because resizing of these images might lead to loss of cellular details which are crucial for detection of the ailment. Similarly, the dataset which we are using has only 374 high-resolution images. To resolve these problems, we applied data augmentation mainly to expand the dataset size and also to tackle the problem of class imbalance. The distribution of lymphoma subtype classes in the dataset before and after applying augmentation techniques is shown in Table 1.

Then we created patches of size 128×128 from each training image such that each image generated 80 patches as shown in Fig. 9 and used these patches to train the model.

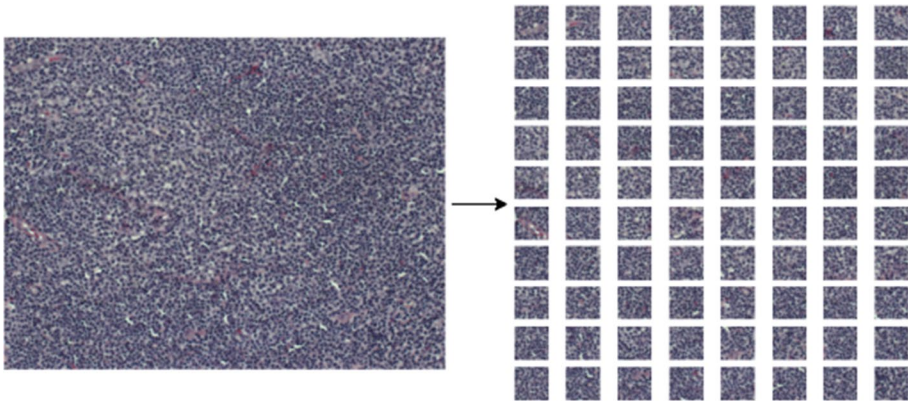


Fig. 9 Image patch creation

5.2 Training and Evaluation

The building block of our model is the fire module. ReducedFireNet in a sense is a variation of SqueezeNet which we have tried to optimize for such medical applications. After experimenting with different numbers of fire modules, we decided to use 4 fire modules applied in sequence with a max-pooling layer following each of the fire modules as using this combination we were able to achieve high accuracy while minimizing the computation requirements and the number of parameters which in turn reduces the FLOPS (Floating Point Operations Per Second) and model size. A dropout layer is also applied before the third fire module to reduce overfitting.

To train and compare our model with other states of the art models, we used stratified K-Fold cross-validation [40]. Cross-Validation is the method of dividing the data in such a manner that every data point will be part of the training and testing data. The Dataset is divided into K different equal parts, (K-1) parts are employed to train the model and the remaining Kth fold is used to test the model, each different Kth fold will be used to test the model in K different iterations. A larger value of K leads to a less biased model. In Stratified Cross-Validation, splitting of the data is done in such a manner that each fold has the same ratio of categorical values so that data distribution across every fold is the same.

We performed 5-Fold cross-validation such that for each time the model trains, 80% (360 samples) of total data is taken as training data and 20% (90 samples) is taken as testing data. Then each of the 360 images was broken into smaller patches with each patch having a size 128×128 . Each image created 80 patches, so finally, the training data had 28,800 (360×80) samples. Using these patches as the training data, we train our model. To evaluate the performance of our model, we utilized the majority voting strategy which selects the prediction with the maximum count among all the patch predictions. Mean F1 scores and accuracies are used for evaluation. In addition to this, we separately report the accuracy (Eq. 2) and F1

Table 2 Fold-wise Accuracy and F1 score of ReducedFireNet model

Metric	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Mean
Accuracy	98.88%	94.44%	97.77%	96.66%	96.66%	96.88%
F1 score	0.988	0.944	0.977	0.966	0.966	0.968

score (Eq. 3) [41] of each fold as shown in Table 2. The F1 score is calculated using Precision (Eq. 4) and Recall (Eq. 5) values.

Accuracy: It denotes the ratio of correct predictions to the total predictions

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \quad (2)$$

Precision: It denotes the ratio of correct positive predictions to all the positive predictions.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3)$$

Recall: It denotes the ratio of correct positive predictions to all the positive observations.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (4)$$

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

$$\text{F1 - Score} = 2 * \frac{\text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}} \quad (5)$$

TP symbolizes true positive, TN represents true negative, FP symbolizes false positive and FN denotes false negative values for multiclass predictions.

6 Results and Limitations

To test the performance of our model, we compared it with state-of-the-art models like ResNet50 [42], Xception [43], and InceptionV3 [44] which are being used for image classification at a large commercial scale. Apart from them we also compared our model with MobileNet [45], which is used for small, low latency, low power image classification for mobile devices.

To show the effectiveness of patch-based training, we have also done the image-based training on each of the models and compared those results with patch-based training. In image-based training, the entire image is resized to a lower size, and the whole image is used for training the model. The results of image-based training are shown in Tables 3 and 4 and the results of patch-based training are shown in Table 5 and Table 6. Tables 3 and 4 report the accuracies and F1 scores obtained when the entire image was used to train the models. Tables 5 and 6 report the accuracies and F1 scores obtained when the image

Table 3 Image based accuracies

Models	Fold 1 (%)	Fold 2 (%)	Fold 3 (%)	Fold 4 (%)	Fold 5 (%)	Mean (%)
ResNet50	47.77	41.11	38.88	72.22	74.44	54.88
InceptionV3	35.55	72.22	51.11	38.88	76.66	54.88
MobileNet	58.88	67.77	68.88	61.11	64.44	64.22
Xception	83.33	68.88	80.00	81.11	73.333	77.33
ReducedFireNet	81.11	52.22	64.44	55.55	60.00	62.66

Table 4 Image based F1 scores

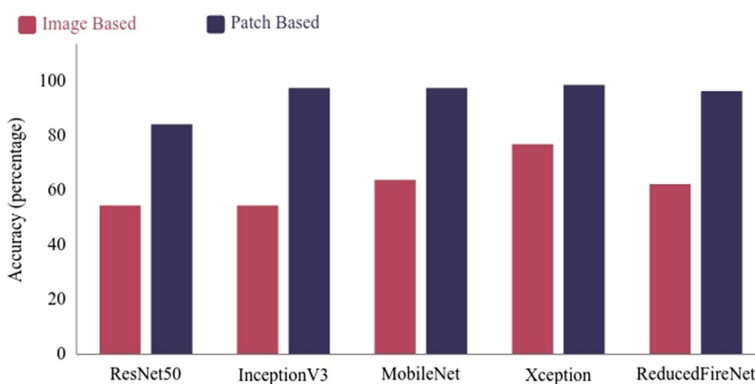
Models	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Mean
ResNet50	0.469	0.401	0.351	0.717	0.738	0.535
InceptionV3	0.280	0.709	0.508	0.286	0.751	0.507
MobileNet	0.573	0.650	0.693	0.604	0.649	0.633
Xception	0.830	0.679	0.798	0.809	0.720	0.767
ReducedFireNet	0.811	0.514	0.604	0.533	0.586	0.609

Table 5 Patch based accuracies

Models	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Mean
ResNet5	93.33%	91.11%	66.66%	81.11%	91.11%	84.66%
InceptionV3	100%	98.88%	93.33%	100%	97.77%	98.00%
MobileNet	98.88%	100%	94.44%	100%	96.66%	98.00%
Xception	96.66%	100%	98.88%	100%	100%	99.11%
ReducedFireNet	98.88%	94.44%	97.77%	96.66%	96.66%	96.88%

Table 6 Patch based F1 scores

Models	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Mean
ResNet50	0.933	0.911	0.636	0.790	0.908	0.836
InceptionV3	1.0	0.988	0.933	1.0	0.977	0.980
MobileNet	0.988	1.0	0.944	1.0	0.966	0.980
Xception	0.966	1.0	0.988	1.0	1.0	0.991
ReducedFireNet	0.988	0.944	0.977	0.966	0.966	0.968

**Fig. 10** Comparison of image-based and patch-based accuracies

was broken down into patches and these patches were used to train the models. In each of these tables, separate accuracy and F1 score for each fold of fivefold cross-validation are reported and their mean value is calculated.

From Fig. 10, it is visible that patch-based training performs much better than image-based training.

These state-of-the-art models have a large number of parameters in their model which in turn leads to a larger model size. Further, these models require a much larger number of floating-point operations per second (FLOPS) in one single instance of the model training.

The following five metrics were employed to analyze the results:

- i. Accuracy
- ii. F1 score
- iii. Number of parameters
- iv. FLOPS
- v. Model size

The experiment results have been summarized in Table 7

From Table 7, we can observe that InceptionV3, Xception, and MobileNet perform very well on both accuracy and F1 score. However, the FLOPS and model size of InceptionV3 and Xception is very high when compared to our proposed ReducedFireNet model. Although the MobileNet requires much fewer FLOPS as compared to InceptionV3 and Xception, the model size of MobileNet remains large. ReducedFireNet performs as well as InceptionV3 and MobileNet with only a 1.12% drop in accuracy while achieving an approximately 100 times smaller model, thereby having a much lesser memory and computational requirements, hence making it a strong candidate for deployment on mobile and IoT devices.

6.1 Compression

To compress our model, we used the process of Quantization, we used the TensorFlow Lite framework to perform Post-Training Quantization to compress our model. TensorFlow Lite (TFlite) is an open-source framework for deep learning, developed by TensorFlow for inference on mobile and IoT devices. Different optimizations are required to allow models to be executed under these constraints, TFlite allows optimizations tailored specifically for hardware accelerations on different kinds of mobile and IoT devices. Post-Training Quantization is the compression technique that compresses a trained TensorFlow model thereby improving the latency for the targeted hardware while having a minimal impact on the accuracy. The TensorFlow model is quantized to its TF Lite format after the process of training has been completed by the model. The entire process is shown in Fig. 11.

Table 7 Detailed comparison of ReducedFireNet against the state-of-the-art models

Models	Accuracy	F1 Score	No. of parameters	FLOPS	Model Size
ResNet50	84.66%	0.836	23,686,019	2.53 G	284.8 MB
InceptionV3	98.00%	0.980	21,827,363	1.47 G	263.0 MB
MobileNet	98.00%	0.980	3,278,019	0.374 G	39.6 MB
Xception	99.11%	0.991	20,959,787	2.96 G	251.8 MB
ReducedFireNet	96.88%	0.968	21,075	0.201 G	0.391 MB

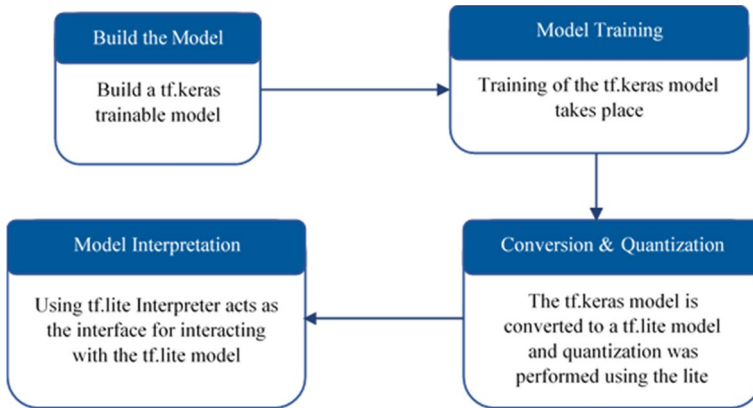
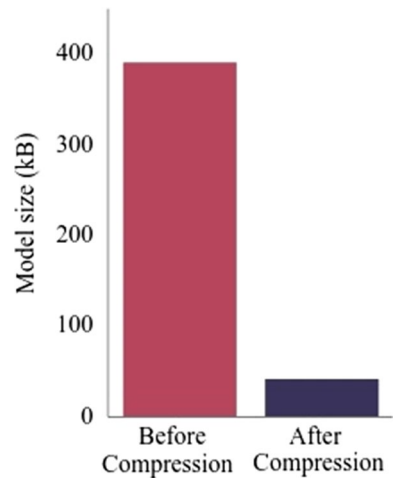


Fig. 11 Steps showing the model compression using TensorFlow Lite

Table 8 Comparison of uncompressed and compressed ReducedFireNet model

Model ReducedFireNet	Accuracy	F1 score	Model Size
Before Compression	96.88%	0.968	391.2 KB
After Compression	95.77%	0.957	42.8 KB

Fig. 12 Comparison of model size before and after compression



As seen in Table 8, performing compression has allowed us to decrease the model size from 391.2 to 42.8 KB with a Moreover, the size of histopathological images is usually large, 1.11% decrease in terms of accuracy. Figure 12 shows the comparison between the size of our proposed model ReducedFireNet with and without compression.

6.2 Limitations

Despite our best efforts to create a precise lightweight model with a low computational requirement for disease detection using histopathological images, our process has few limitations. Firstly data augmentation methods performed on the dataset could be further improved by generating high-quality synthetic data using a variation of GANs and employing techniques like texture transfer and style transfer. Regarding the compression of our model, we have utilized post-training quantization, which could further be improved slightly by utilizing an efficient pruning strategy for our proposed model.

7 Conclusion and Future Work

The importance of IoMT for providing effective, low-cost, and timely medical care to patients is undeniable. To aid this medical facility, early diagnosis of critical diseases is equally important. Timely detection of diseases can save millions of lives. However, accurate and spontaneous disease prognosis task suffers from several issues. The transfer of real-time patient data to the next IoT level, for computation, can cause a delay in the decision. Moreover, the size of histopathological images is usually large, which will require large bandwidth for data transfer. These issues can be resolved if the collected data is analyzed locally at the IoMT devices. To resolve these issues, we proposed a ReducedFireNet model, which is a high performing, low weight model comparable with state-of-the-art models while maintaining its small size and fewer FLOPS requirements. However, the model size remained fairly sizeable and the compression process allowed us to decrease its size to 43 KB with only a minute drop in terms of accuracy.

For our future work related to this paper, we will work on mitigating our process's limitations by utilizing a CycleGAN to develop high-quality synthetic histopathological images to further augment our dataset and devising an effective strategy for pruning to further compress our proposed ReducedFireNet model. Apart from this we also want to work on making other medical applications of deep learning such as detection of cardiac arrhythmia and image segmentation used for nuclei segmentation, etc. feasible on IoMT devices to further improve medical pipelines.

Acknowledgements This work was partially supported by the National Natural Science Foundation of China (Grant No. 61872084); Guangdong-Hong Kong-Macao Joint Laboratory for Intelligent Micro-Nano Optoelectronic Technology (No. 2020B1212030010).

References

1. La HJ, Jung HT, Kim S (2015) Extensible disease diagnosis cloud platform with medical sensors and IoT Devices. 2015 3rd International Conference on Future Internet of Things and Cloud, 371–378
2. Hassanaliereagh, Moeen, Alex Page, Tolga Soyata, Gaurav Sharma, Mehmet Aktas, Gonzalo Mateos, Burak Kantarci, and Silvana Andreescu. "Health monitoring and management using Internet-of-Things (IoT) sensing with cloud-based processing: Opportunities and challenges." In 2015 IEEE International Conference on Services Computing, pp. 285–292. IEEE, 2015
3. Tyagi, Sapna, Amit Agarwal, and Piyush Maheshwari. "A conceptual framework for IoT-based health-care system using cloud computing." In 2016 6th International Conference-Cloud System and Big Data Engineering (Confluence), pp. 503–507. IEEE, 2016

4. Lakhan A, Mastoi Q-U-A, Elhoseny M, Memon MS, Mohammed MA (2021) Deep neural network-based application partitioning and scheduling for hospitals and medical enterprises using IoT assisted mobile fog cloud. *Enterp Inf Syst* . <https://doi.org/10.1080/17517575.2021.1883122>
5. Mendez AJ, Tahoces PG, Lado MJ, Souto M, Vidal JJ (1998) Computer-aided diagnosis: automatic detection of malignant masses in digitized mammograms. *Med Phys* 25:957–64 (**PubMed: 9650186**)
6. Goodfellow I, Bengio Y, Courville A (2016) *Deep Learning*, Cambridge, MA. MIT Press, USA
7. Tao X, Gao H, Shen X, Wang J, Jia J (2018) "Scale-Recurrent Network for Deep Image Deblurring," *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, 2018, pp. 8174–8182, doi: <https://doi.org/10.1109/CVPR.2018.00853>
8. Rubin R, Strayer D, Rubin E, McDonald J (2007) *Rubin's pathology: clinicopathologic foundations of medicine*. Lippincott Williams & Wilkins, USA
9. Misra D, Crispim-Junior C, Tougne L (2020) Patch-based CNN evaluation for bark classification. *hal-02969811v2*
10. Han, Song & Mao, Huizi & Dally, William. (2016). *Deep compression: compressing deep neural networks with pruning, trained quantization and Huffman coding*
11. Sun X, Qian W, Song D (2004) Ipsilateral-mammogram computer-aided detection of breast cancer. *Comput Med Imag Graph* 28(3):151–158
12. Kumar Y, Aggarwal A, Tiwari S, Singh K (2018) An efficient and robust approach for biomedical image retrieval using Zernike moments. *Biomed Signal Process Control* 39:459–473
13. Saidin N, Ngah UK, Sakim HAM, Ding NS, Hoe MK, Shuaib IL (2010) "Density based breast segmentation for mammograms using graph cut and seed based region growing techniques," in *Proc. 22nd Int. Conf. Comput. Res. Develop.*, 2010
14. Xu S, Liu H, Song E (2011) Marker-controlled watershed for lesion segmentation in mammograms. *J Digit Imag* 24(5):754–763
15. Lahoura V, Singh H, Aggarwal A, Sharma B, Mohammed MA, Damaševičius R, Kadry S, Cengiz K (2021) Cloud computing-based framework for breast cancer diagnosis using extreme learning machine. *Diagnostics* 11:241. <https://doi.org/10.3390/diagnostics11020241>
16. Huang Q, Zhang F, Li X (2018) Machine learning in ultrasound computer aided diagnostic systems: A survey. *BioMed Res Int* 2018:1–10
17. Doyle S, Hwang M, Shah K, Madabhushi A, Feldman M, Tomaszewski J (2007) Automated grading of prostate cancer using architectural and textural image features. In: *2007 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro, 2007*, pp 1284–1287. <https://doi.org/10.1109/ISBI.2007.357094>
18. Kim B, Kim KC, Park Y, Kwon JY, Jang J, Seo JK (2018) Machine learning- based automatic identification of fetal abdominal circumference from ultrasound images. *Physiol. Meas.* 39(10):105007
19. Jafari-Khouzani K, Soltanian-Zadeh H (2003) Multiwavelet grading of pathological images of prostate. *IEEE Trans Biomed Eng* 50:697–704
20. Rajpoot N, Arif M, Bhalerao, A (2007) *Proceedings. 2007. Unsupervised Learning of Shape Manifolds*
21. Madabhushi A, Feldman M, Metaxas D, Tomaszewski J, Chute D (2005) Automated detection of prostatic adenocarcinoma from high resolution Ex Vivo MRI. *IEEE Trans Med Imag* 24:1611–1625
22. Keenan S, Diamond J, Glenn McCluggage W, Bharucha H, Thompson D, Bartels P, Hamilton P (2001) An automated machine vision system for the histological grading of cervical intraepithelial neoplasia (CIN). *J Pathol* 192(3):351–362
23. Roy T, Sirohi N, Patle A (2015) "Classification of lung image and nodule detection using fuzzy inference system." *International Conference On Computing, Communication & Automation*. DOI: <https://doi.org/10.1109/CCAA.2015.7148560>
24. Orlov NV, Chen WW, Eckley DM, Macura TJ, Shamir L, Jaffe ES, Goldberg IG (2010) Automatic classification of lymphoma images with transform-based global features. *IEEE Trans. Inf. Technol. Biomed.* 14(4):1003–1013
25. Nascimento MZd, Neves L, Duarte SC, Duarte YAS, Batista VR (2015) Classification of histological images based on the stationary wavelet transform, in: *Journal of Physics: Conference Series*, vol. 574, IOP Publishing, 2015, p. 012133
26. Nascimento MZD, Martins AS, Tosta TAA, Neves LA (2018) Lymphoma images analysis using morphological and non-morphological descriptors for classification. *Comput Methods Progr Biomed* 163:65–77
27. Abdulkareem KH et al (2021) Realizing an effective COVID-19 diagnosis system based on machine learning and IOT in smart hospital environment. *IEEE Internet Things J*. <https://doi.org/10.1109/JIOT.2021.3050775>

28. Javed Awan M, Mohd Rahim M, Salim N et al (2021) Efficient detection of knee anterior cruciate ligament from magnetic resonance imaging using deep learning approach. *Diagnostics* 11:105. <https://doi.org/10.3390/diagnostics11010105>
29. Al-Waisy AS, Mohammed MA, Al-Fahdawi S, Maashi MS, Garcia-Zapirain B et al (2021) Covid-deepnet: hybrid multimodal deep learning system for improving covid-19 pneumonia detection in chest x-ray images. *Comput, Mater Continua* 67(2):2409–2429
30. Mikołajczyk A, Grochowski M (2018) Data augmentation for improving deep learning in image classification problem. In: 2018 International Interdisciplinary PhD Workshop (IIPHDW), 2018, pp 117–122. <https://doi.org/10.1109/IIPHDW.2018.8388338>
31. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial networks. *Adv Neural Inf Process Syst*. <https://doi.org/10.1145/3422622>
32. Iandola, Forrest & Han, Song & Moskewicz, Matthew & Ashraf, Khalid & Dally, William & Keutzer, Kurt. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size
33. Krizhevsky A, Sutskever I, Hinton G (2012) ImageNet classification with deep convolutional neural networks. *Neural Inf Process Syst*. <https://doi.org/10.1145/3065386>
34. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res* 15(1):1929–1958
35. Netron: Visualizer for neural network, deep learning and machine learning models. <https://www.lutzroeder.com/ai>
36. Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Rafal Jozefowicz, Yangqing Jia, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Mike Schuster, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org
37. Chollet, F., & others. (2015). Keras. GitHub. Retrieved from <https://github.com/fchollet/keras>
38. Orlov N, Chen W, Eckley D, Macura T, Shamir L, Jaffe E, Goldberg I (2010) Automatic classification of lymphoma images with transform-based global features. *IEEE Trans Inf Technol Biomed: a Pub IEEE Eng Med Biol Soc* 14:1003–1013. <https://doi.org/10.1109/TTTB.2010.2050695>
39. Deng J, Dong W, Socher R, Li L, Li K, Li Fei-Fei (2009) "ImageNet: A large-scale hierarchical image database," 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, 2009, pp. 248–255, doi: <https://doi.org/10.1109/CVPR.2009.5206848>
40. Hastie T, Tibshirani R, Friedman J (2001) *The Elements of Statistical Learning*. Springer, New York
41. Powers DMW (2011) Evaluation: from precision, recall and F-Measure to ROC, informedness, markedness & correlation. *J Mach Learn Technol* 2(1):37–63
42. He, Kaiming & Zhang, Xiangyu & Ren, Shaoqing & Sun, Jian. (2016). Deep residual learning for image recognition. 770–778. <https://doi.org/10.1109/CVPR.2016.90>
43. Chollet F (2017) Xception: Deep Learning with Depthwise Separable Convolutions. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp 1800–1807. <https://doi.org/10.1109/CVPR.2017.195>
44. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2016) Rethinking the Inception Architecture for Computer Vision. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 2818–2826. <https://doi.org/10.1109/CVPR.2016.308>
45. Howard AZ, Chen M, Kalenichenko B, Wang D, Weyand W, Andreetto T, Adam M (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications

Authors and Affiliations

Koyel Datta Gupta¹ · Deepak Kumar Sharma²  · Shakib Ahmed² · Harsh Gupta² · Deepak Gupta³ · Ching-Hsien Hsu^{4,5,6}

Koyel Datta Gupta
koyel.dg@msit.in

Shakib Ahmed
shakiba.it18@nsut.ac.in

Harsh Gupta
harshg.it18@nsut.ac.in

Deepak Gupta
deepakgupta@mait.ac.in

Ching-Hsien Hsu
robertchh@asia.edu.tw

- ¹ Department of Computer Science and Engineering, Maharaja Surajmal Institute of Technology, New Delhi, India
- ² Department of Information Technology, Netaji Subhas University of Technology, New Delhi, India
- ³ Department of Computer Science and Engineering, Maharaja Agrasen Institute of Technology, Delhi, India
- ⁴ Guangdong-Hong Kong-Macao Joint Laboratory for Intelligent Micro-Nano Optoelectronic Technology, School of Mathematics and Big Data, Foshan University, Foshan 528000, China
- ⁵ Department of Computer Science and Information Engineering, Asia University, Wufeng, Taiwan
- ⁶ Department of Medical Research, China Medical University Hospital, China Medical University, Taichung, Taiwan