FOREX Publication
Open Access | Rapid and quality publishing

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 3 | Pages 694-704 | e-ISSN: 2347-470X

Revaluating Pretraining in Small Size Training Sample Regime

Vandana Khobragade[1], Jagannath Nirmal[2], and Shreyansh Chedda[3]

[1]*Department of Electronic and Telecommunication, LTCE, University of Mumbai, Mumbai, India, vandanakhobragade@ltce.in*
[2]*Department of Electronics, KJSCE, Mumbai, India, jhnirmal@somaiya.edu*
[3]*Department of Computer Science, VJTI, Mumbai, India, shreyansh.chheda@gmail.com*

***Correspondence:** Vandana Khobragade; Email: vandanakhobragade@ltce.in

ABSTRACT- Deep neural network (DNN) based models are highly acclaimed in medical image classification. The existing DNN architectures are claimed to be at the forefront of image classification. These models require very large datasets to classify the images with a high level of accuracy. However, fail to perform when trained on datasets of small size. Low accuracy and overfitting are the problems observed when medical datasets of small sizes are used to train a classifier using deep learning models such as Convolutional Neural Networks (CNN). These existing methods and models either always overfit when training on these small datasets or will result in classification accuracy which tends towards randomness. This issue stands even when using Transfer Learning (TL), the current standard for such a scenario. In this paper, we have tested several models including ResNet and VGGs along with more modern models like MobileNets on different medical datasets with transfer learning and without transfer learning. We have proposed solid theories as to why there exists a need for a more novel approach to this issue, and how the current methodologies fail when applied to the aforementioned datasets. Larger, more complex models are not able to converge for smaller datasets. Smaller models with less complexity perform better on the same dataset than their larger model counterparts.

Keywords: Acute Lymphoblastic Leukemia, Convolutional Neural Network, Deep Neural Network, Transfer Learning, White Blood Cells.

ARTICLE INFORMATION

Author(s): Vandana Khobragade, Jagannath Nirmal, and Shreyansh Chedda;
Received:20/06/2022; **Accepted**: 13/09/2022; **Published**: 22/09/2022;
E- ISSN: 2347-470X;
Paper Id: IJEER220626;
Citation: 10.37391/IJEER.100346
Webpage-link:
https://ijeer.forexjournal.co.in/archive/volume-10/ijeer-100346.html

Publisher's Note: FOREX Publication stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. INTRODUCTION

Investigation of chronic blood-related diseases like Leukemia, Erythrocytosis, and Anemia discussed in [1] and [2], involves microscopic examination of blood. Many critical blood-related diseases can be controlled by examining the morphology of blood cells. The analysis involves the study of morphological features of blood cells. One such disease related to abnormal growth of WBC (Leukocytes) is Leukemia. Here, the work is towards analyzing a small-size data set from a specific type of leukemic blood smear image. The type of leukemia under consideration is Acute Lymphoblastic Leukemia (ALL). In this kind of abnormality, white blood cells (WBC) are not developed. 'Lymphoblasts' is the name given to such underdeveloped cells. These blasts block the production of robust red blood cells, platelets, and even fully developed white blood cells (Leukocytes). ALL is usually diagnosed by taking a complete blood count. In this test, the doctor will check if the number of WBCs has increased and whether there are signs of leukemia cells or not. Sometimes these symptoms are not enough for doctors to endorse that the patient has leukemia.

However, to date, the morphologic profile of blood cells is primarily based on handling blood smear slides manually. In [3] it is discussed that such investigation involves visual examination of slides that lacks quality control and economic scalability. As stated in [4], it is believed that the preparation and analysis of blood smears require highly qualified technicians otherwise, the analysis may give erroneous results, which are not acceptable in the field of medical diagnosis. To overcome the limitations mentioned above, studies have presented different methods of computer-aided diagnosis for ALL. In these methods, microscopic images of blood are analyzed to detect Leukemia. These methods are more competent, faster, cost-effective, and accurate than manual methods. To implement automation in the microscopic examination, firstly with the help of some image acquisition techniques, microscopic images are captured and stored as digital files. These digital images are fed to a structured program, also known as an algorithm to extract meaningful data from it. The program must find and recognize patterns in the data, in an efficient manner. A computer program (algorithm) designed to find patterns in structured data is attributed to machine learning (ML). In [5] utilizing ML methodologies, a detailed review of current advancements in brain tumor detection is presented.

Image processing and machine learning techniques are predominantly used in the automated or computer-aided analysis of ALL-like diseases. Preprocessing, image

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 3 | Pages 694-704 | e-ISSN: 2347-470X

segmentation, feature extraction, and classification are the basic steps involved in image processing based on medical image analysis tasks. The success of each step is dependent upon the successful completion of the previous step. High classification accuracy is achievable, provided all the above-mentioned steps are completed. As discussed in [6], a new modified approach is proposed for brain tumor classification as well as feature extraction from Magnetic Resonance Imaging (MRI). The study found that these steps are essential and are problem-dependent.

Recently, deep learning techniques have become promising options for the analysis of medical images. For example, work in [7] and [8] reported that convolutional neural networks have been applied as a microscopic analysis methodology. Designing and training a deep neural network (DNN) is an important but lengthy process. The transfer learning approach has been explored as a solution to this problem of training from scratch.

In this work, CNN's Selftrained models and the transfer learning fundamentals are applied. In transfer learning, DNN that has successfully tackled one problem is utilized and tuned to solve another task. In deep learning, CNN predominantly takes care of visual image analysis. In the 1980s CNN was invented and adopted a breakthrough in the year 2000 with the introduction and popularity of Graphics Processing Units (GPUs). Dan, in 2011 introduced different types of CNN, which were based on high-performance GPUs. Survey paper [9], discussed the history and evolution of GPU hardware architecture. In [10], Ciresan has analyzed that the use of high-performance GPU has helped in achieving a higher recognition test error rate of 0.35% during digit recognition (MNIST). Also, 3D recognition of object (NORB) and natural images (CIFAR10) detection gave recognition test error rates of 2.53% and 19.51%, respectively. In 2012, The Krizhevsky model won the ImageNet large-scale visual recognition challenge. This model, as discussed in [11], [12], and [13], was trained on 1.2 million high-resolution images with fine details. The model consists of 8 weighted layers, out of which 5 are convolution layers, and 3 are fully connected layers. ReLU, as described in [14], is used as an activation function. Image identification has been achieved by a few deep models like VGG, MobileNet, and DenseNet. These models are trained on ImageNet datasets for the task of identifying images. A lightweight deep neural network, created using optimized architecture, is based on the concept of depth-wise separable convolution. MobileNet as reported in [15], is an example of one such architecture which utilizes factorized convolution.

Generally, in a standard convolution process, all input channels undergo filtering operation simultaneously and output is decided based on all combined values. However, to reduce the computational complexity experienced in standard convolution another approach is used. In MobileNet, depth-wise convolution and pointwise convolution are considered. In depth-wise convolution, individual channel experiences convolution operation separately. These individual outputs obtained from all the channels of the depth-wise stage are linearly combined using 1x1 pointwise convolution.
A convolution layer with n number of filters (kernels) each of size D_F xD_Fx1 is considered. The image of size D_I x D_I x m undergoes convolution to produce a result of D_R x D_R x n. The computational cost (Cost) for standard convolution is estimated as

$$\text{Cost} = D^2F \times D^2R \times m \times n \quad (1)$$

Depth-wise Convolution: Here, in one convolution operation the number of multiplications per kernel is given by the following equation.

$$M_{dF1} = D^2F \quad (2)$$

Pointwise Convolution: Here, kernels of size 1×1×m are used to linearly combine the outputs of all the channels of depthwise convolution. Hence, the number of multiplications per kernel on one convolution operation is:

$$M_{pF1} = m \quad (3)$$

According to [16], an Open-source system called DenseNet computes dense, multiscale features from the convolutional layers of a CNN-based object classifier. The above discussion about different architecture reveals the fact that deep CNN can lead to excellent performance.

In the case of training models like VGG, MobileNet, and DenseNet using CNNs, giant ImageNet datasets are mainly used. In reality, medical imaging research lacks the availability of massive datasets like ImageNet. Collecting a large domain-dependent set of data can be expensive for a specific task. A technology that can methodically run a deep neural network for smaller datasets without compromising on high performance is a challenge. As a solution to the above problem, transfer learning is explored in the case of training the data that belongs to the category of “smaller datasets”. Obtaining the pre-trained model weights, that were developed for one task and reusing them on another task without learning from scratch, is the core concept behind transfer learning. So, transfer learning can be looked upon as an approach that will reduce the optimizations, achieving lesser training time and improved performance.

However, experiments carried out in this work show that the implementation of pre-trained models always does not prove efficient. Sometimes, these models fail to achieve good results. In this work, the Self-training criteria are explored as a counteraction to pre-training. The experimental results highlight the fact that the performance parameters obtained for small-size datasets trained with the help of random weights are comparable with its counterpart in which Image Net retraining is used. Even large models like ResNet101, when trained from scratch using a small set of data (ALLIDB2), yield comparable results without showing any overfitting. The rest of the paper is arranged as follows. The succeeding section describes the related works in which traditional and recent methods are discussed. Next, the methodology implemented in this work is discussed. This is followed by a section on the experimental

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 3 | Pages 694-704 | e-ISSN: 2347-470X

setting in which algorithms and experimentation are elaborated. The fifth section is all about the result and its analysis. It is followed by a section that concludes the paper.

2. RELATED WORK

Research conducted in the field of infirmity, specifically blood disease diagnosis and detection, are discussed in this section. Related traditional methods examined in [17], [18], and [19], consider conventional image processing steps like preprocessing, image segmentation, feature extraction, and classification. Deep learning methods, which utilizes extensive learning task through the deep neural network are examined in [20], [7], and [8].

2.1 Traditional Method

A cloud-assisted mobile framework that classifies WBC into five classes is discussed in [17]. The color k-mean segmentation algorithm is adopted by the framework to segment the leukocyte or white blood cells (WBCs). In this work, the segmentation algorithm employed various morphological operations to remove unrelated components. The principal component analysis method is adopted to extract varied geometric, statistical, and textural features from segmented WBCs. Finally, a multiclass support vector machine (SVM) is employed for classification purposes. This framework utilizes a dataset of 1030 blood smear images, and successfully produced an average accuracy of 98.6%.

Conventional medical image processing steps like preprocessing, nucleus segmentation, feature extrication, feature selection, and classification are incorporated into the system suggested by [18]. For segmentation, methods like ellipse curve fitting, morphological operations, and thresholding are used in [18]. Features are extracted from the nucleus and cytoplasm region. The greedy search algorithm is utilized to decide upon selected features. Finally, Naive Bayes and linear classifiers using selected features are employed for classification.

2.2 Transfer Learning

Training an image classifier that will achieve near-human-level accuracy in image classification is preferred by many researchers. Such classification requires a massive amount of data, lots of time in hand, and considerable computational power. In deep learning, one of the popular training techniques is transfer learning, where models that have been trained for a specific task are reused as a starting point for another model that is designated to perform some different tasks. The goal of transfer learning is to improve the learning skills in the target task provided the knowledge comes from the source task [21]. Models trained on large image datasets like ImageNet, and COCO are built by some researchers, and later, they shared the models for reuse. Transfer of knowledge would improve learning performance and reduce the effort of data collection [22].

According to work presented in [23] and [24], object classification results are significantly improved by using transfer learning concepts. However, as recorded in [25], when a new classifier is trained with a smaller number of samples, there exists a risk of overfitting the new data, leading to poor generalization. To overcome this problem, Shrivastava et al. [26] came up with the concept of dropout. Also, the concept of data augmentation was reported in [27], [28], and [29]. Above mentioned techniques are proven useful in preventing the neural network from overfitting. In [30] deep learning is explored to classify images based on large-scale datasets. Deep neural networks utilizing the concept of transfer learning is being explored for the classification of medical image, and the same is discussed in [31] and [32]. The convolutional neural network architecture with dropout and fine-tuning strategy used for transfer learned representation of huge image classification dataset is proposed in [33], [34], [35], and [36]. In this paper, the prime focus is on utilizing the pre-trained DNN models and also on models trained from scratch in a minimal dataset regime.

2.3 Deep Learning

Multilayer DNN is implemented either by using the supervised method, as recorded by [37], or using unsupervised methods, as supported by [38] and [39]. Work analyzed in [40] uses dropout parameters in the deep model to attain better performance. Similar to our proposed work the study in [41] has experimented with nine pre-trained CNN models. Their work was aimed toward improving diagnostic efficiency for COVID-19 detection and establishing comparative analysis of different pre-trained CNN models in a limited class CT image dataset. In [42], AlexNet is used for both feature extraction and classification with a dataset consisting of 2820 images. Habibzadeh et al. in [43] implemented a work classifying WBCs using deep learning models into four primary types including Neutrophils, Eosinophils, Lymphocytes, and Monocytes. To analyze the accuracy, two fine-tuned versions of ResNets are explored in that work. Deep learning-based ExceptionNet under the transfer learning paradigm is discussed in [44], here the proposed model uses chest-X rays collected from the open-source dataset (COVID -19 Dataset) using K-10 cross-validation.

Here, the proposed work is mainly focused on exploring Selftrained and Pretrained versions of VGG, MobileNet, and ResNet. Different from the analysis of [40], in the present work, the data-augmentation transformations along with dropout are being added to the data pre-processing configuration. However, the above-mentioned pre-trained models fail to achieve the expected results. The objective of the proposed work is to present the argument through experimentation that even appropriately modified pre-trained CNNs having depth, fail to correspond to a small dataset.

3. METHODOLOGY

3.1 Framework of approach

In the controlled experiments performed to ablate the role of ImageNet pretraining, architectural improvements are not the prime concern. It is desired to understand the impact of ImageNet pre-trained models, which are trained from scratch for a different domain task. Experiments were carried out on different kinds of models, namely MobileNet, ResNet, and VGGs. These were tested with pre-training and self-training

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 3 | Pages 694-704 | e-ISSN: 2347-470X

on different datasets. As shown in *figure 1* the image is preprocessed and goes through the stage of data augmentation. Both pretrained and self-trained above-mentioned models are explored in this experimentation. Output in the form of accuracy is analyzed.

3.2 Data description

In the medical domain, it is common that images usually originate from some sort of medical imaging devices like an MRI scan or ultrasound scan. As discussed in [45], the images in ALL-IDB1 and ALL-IDB2 are photographs of biological cells or membranes as seen under a microscope. The dataset referred to as the ALL-IDB2 dataset (**Dataset1)**, contains 260 cell images: 130 normal and 130 affected by Acute Lymphoblastic Leukemia. This image set has been designed for testing the performances of classification systems. The ALL-IDB2 is a collection of cropped areas of interest of normal and blast cells that belong to the ALL-IDB1 dataset. It contains 260 images, and 50% of these represent lymphoblasts. ALL-IDB2 images have similar gray-level properties to the images of the ALL-IDB1, except for the image dimensions. The dataset is public and freely available. The ALL-IDB2 image files are named with the notation ImXXX_Y.jpg where XXX is a progressive 3-digit integer and Y is a boolean digit equal to 0 if the cell placed in the center of the image is not a blast cell, and equal to 1 if the cell that is placed in the center of the image is a blast cell. Please note that all images labeled with Y=0 are from healthy individuals, and all images labeled with Y=1 are from ALL patients.

Another dataset that is used to test the performance of models is the CNMC dataset (**Dataset2)**. As discussed in [46], CNMC is a dataset of cells with labels (normal versus cancer). It is provided to train machine learning-based classifiers to identify normal cells from leukemic blasts (malignant/cancer cells). These cells are segmented from the microscopic images. These images are representative of images in the real world because they contain some staining noise and illumination errors. However, these errors have been mainly fixed by the method of stain color normalization. This dataset was also used for IEEE ISBI 2019 conference challenge: Classification of Normal vs. Malignant Cells in B-ALL White Blood Cancer Microscopic Images.

3.3 Data augmentation

If a smaller number of training cases are used then there is a chance of overfitting. To reduce the problem of overfitting, a couple of strategies such as data augmentation and dropout need to be implemented. To make the most out of a few training examples, "augment" the available training samples via several random transformations so that at training time, the model will have a broader and more robust distribution so that it learns the best features from the dataset. This helps prevent overfitting and helps the model generalize better. The transformations used are random-crop, horizontal flip, and normalize. After applying the transformations to a benign image from the training set of ALLIDB2, random variants produced are seen in *figure 2* below. A random image is chosen from the training data to anticipate the intermediary results. Next, a figure is generated in which each row indicates the output of that layer, and every image in that row indicates a specific filter that is present in the output feature map. *Figure 3* shows the conversion of image pixels to an increasingly abstruse and compact depiction.

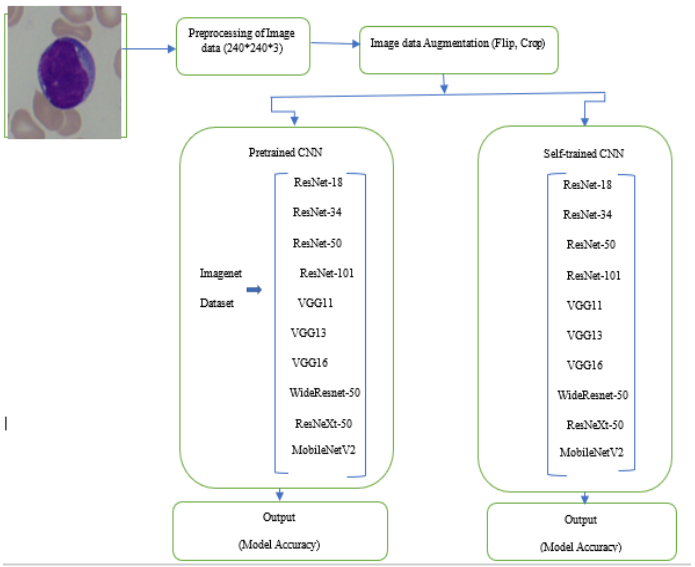

Figure 1: Framework of proposed work

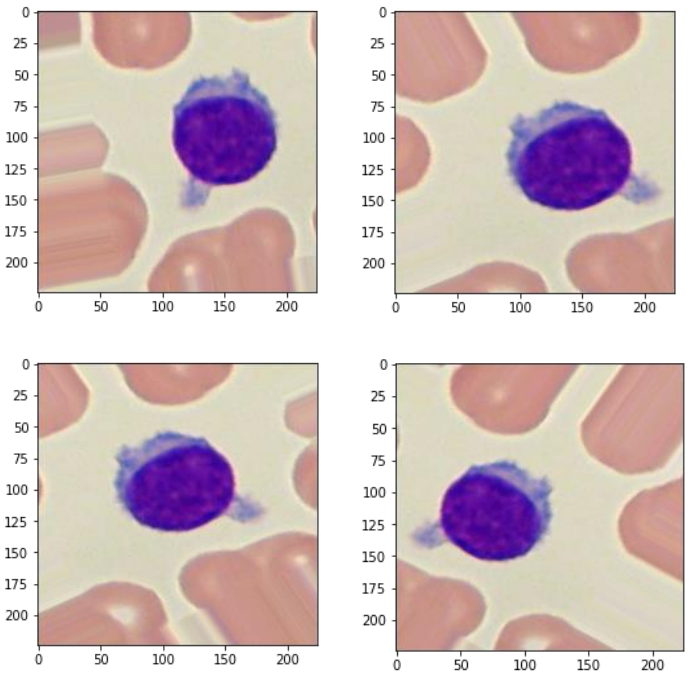

Figure 2: Image augmentation

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 3 | Pages 694-704 | e-ISSN: 2347-470X

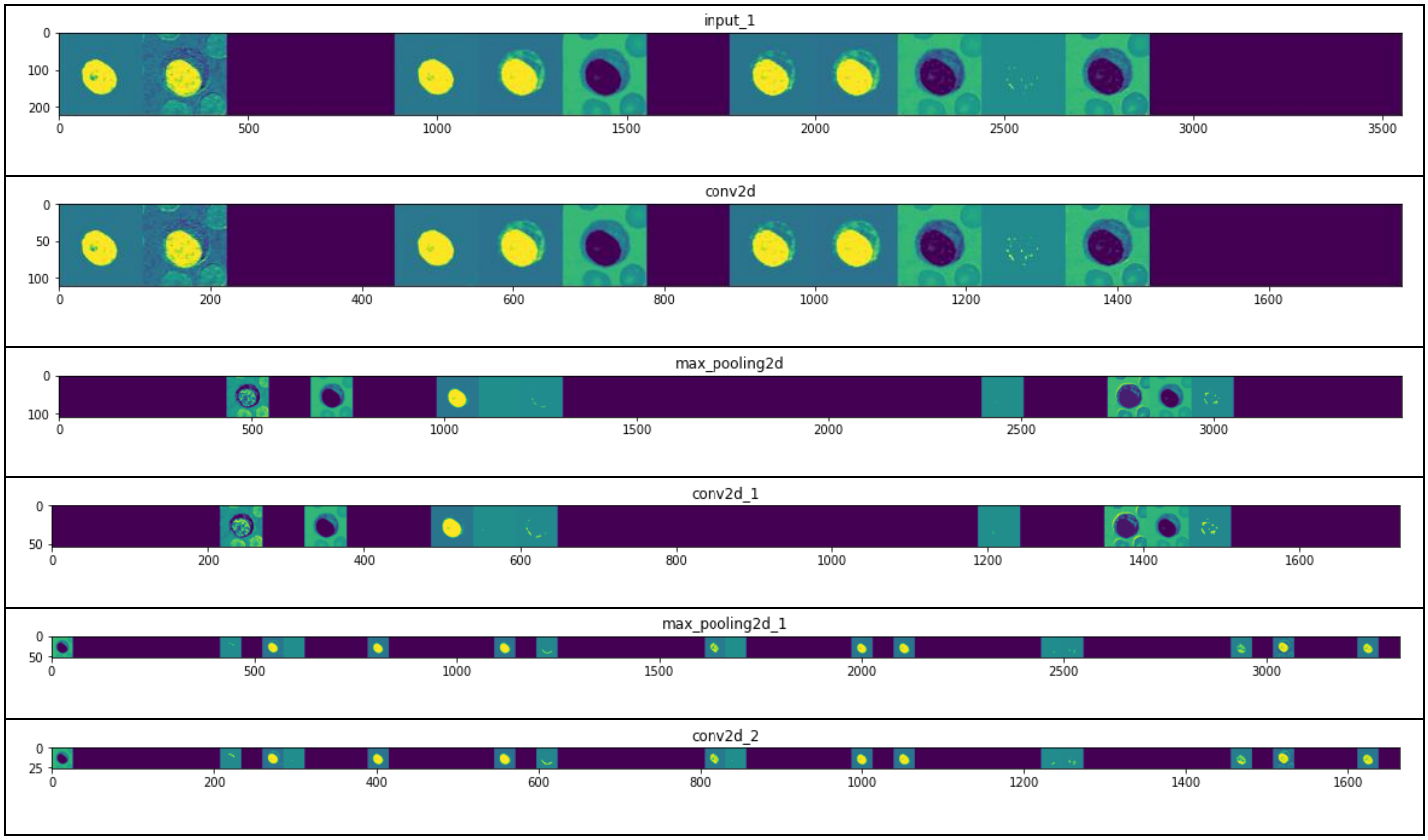

Figure 3: Intermediate results in CNN

4. EXPERIMENTAL SETTINGS

The experiments performed in this work aim towards figuring out "how helpful are pre-trained models for medical datasets". The shapes and outlines of images used in pretrained models are different from those in medical datasets. For this reason, in medical image classification tasks, pre-trained models that are trained from scratch should be used. Further, it is observed that medical datasets are usually very small in size and bigger models tend to overfit the data. This is where smaller models are more helpful as they respond better to smaller datasets. At the same time, they are faster and more efficient for deployment on mobile devices with lesser computational power. The algorithm is discussed below and the flowchart is shown in *figure 4*.

4.1 Algorithm of proposed work

Import all dependencies and datasets from the file system
-Formulate image transformation with resizing, random horizontal flip, and Normalization methods to improve dataset robustness.
-Set parameters for learning rate, batch size, epochs, etc.
-Load data using torch data loader
-Set model to be tested with pre-training or no pretraining options.
-Set optimizer with stochastic gradient descent
-Start model training
-Save model accuracy, time for running training epoch, and time for running test epoch
-Plot graphs for accuracy, training time and testing time
-Save and compare different models and build graphs accordingly.

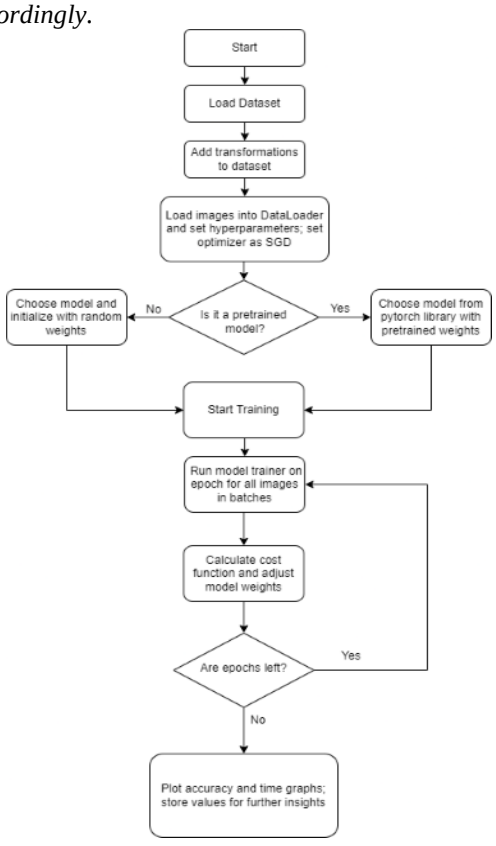

Figure 4: Flowchart of the Algorithm

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 3 | Pages 694-704 | e-ISSN: 2347-470X

4.2 Experimentation

In these experiments, the medical imaging task of investigating Leukemia (ALL) is taken into account. In this regard deep neural network architectures, including both standard architectures generally used for medical imaging (ResNet50 and VGG16) as well as their smaller, less complex, lighter versions, are explored. One more modern model meant for mobile devices, MobileNetV2 is also tested.

For each network architecture, either in pretraining or self-training cases, a Stochastic Gradient Decent (SGD) optimizer in a maximum of 240 epochs is used. Batch size is set to 8 for test and train both with an initial learning rate of 0.05. SGD is an optimization algorithm that evaluates the gradient of error for the existing state of the model utilizing the samples from the training dataset. It is followed by updating the weights of the models using the backpropagation algorithm. The extent to which the weights are updated during training depends on the learning rate, given by *equation (4).*

$$\omega^i \leftarrow \omega^i - \eta \frac{\partial \mathcal{L}}{\partial \omega^i} \tag{4}$$

ω^i = weights in layer i,

$\frac{\partial \mathcal{L}}{\partial \omega^i}$ = partial derivative of the loss function with reference to the weights ω of the model,

η = learning rate.

Equation 4 expresses the modification of parameters according to SGD. Here, ω^i in LHS and RHS of ← indicates the updated and old weights in layer i respectively. $\frac{\partial \mathcal{L}}{\partial \omega^i}$ indicates the partial derivative of the loss function with reference to the weights ω of the model. Proportional to the learning rate η, the weights ω^i are updated in a direction that is negative of the gradient. This is done to minimize the loss function. The rule of updating the weights is called Gradient Descent. This is exemplified in Figure 5 the term 'G' is used to denote the gradient. Each round of parameter updates requires passing a batch of examples through the NN, averaging their loss, and accumulating gradients via backpropagation. These steps constitute one iteration of stochastic gradient descent (SGD). It can be concluded from the above discussion that SGD is nothing but updating the parameters with the help of the gradient estimated from the batch of training examples.

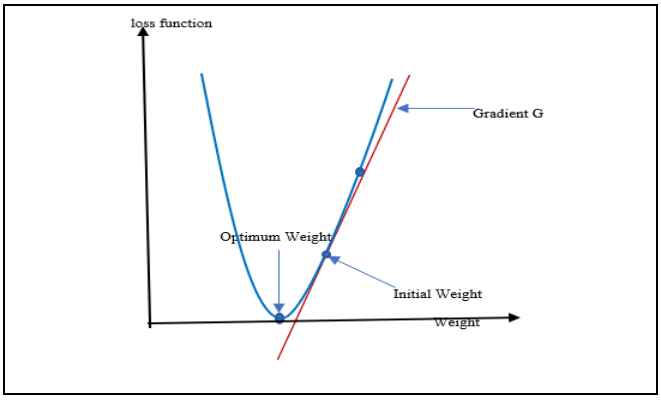

Figure 5: Illustration of the gradient descent

In an experiment with VGG11 and VGG 13 on the CNMC data set, it is observed that the models were not converging and were giving random outputs. One of the solutions to this problem was reducing the learning rate. The learning rate was reduced from 0.05 to 0.01 in a staggered manner and it is found that at 0.01 the self-trained VGG models achieved convergence, but the pre-trained model still refused to converge. Next, in experimentation, it was observed that the ResNet are small in "size" and were able to converge under similar conditions with all models. Later it was also noticed that MobileNets needed a lower learning rate and adopted the learning rate of 0.01 during all the experiments.

For experimentation purposes, the same learning rate values were utilized between all the models to draw an equal and fair conclusion (whether pre-training helps or does not help). Python is used as a programming language to train CNN models. The images were re-sampled to 224 × 224-pixel resolutions to suit the input requirements of customized and pre-trained CNNs. These are normalized to facilitate faster convergence. The models were trained and tested on a DGX Station, Ubuntu OS, with Intel® Xeon® CPU E5-2698v4 2.2-GHz processor (20-core), NVIDIA Tesla® V100-DGXS-32GB with 32 GB per GPU (128 GB total) of GPU memory. Each of CNN's architecture utilizes some hyper-parameters for deep transfer learning. Parameter values are presented in *table 1*.

5. RESULT and ANALYSIS

5.1 Result Analysis

As discussed in the dataset description *section 3.2*, two publicly available datasets are used to perform the experimental study. Accuracy is used as a performance metric to examine the efficacy of the classification model. This accuracy measure will work, as the datasets under consideration contain an equal number of images for both classes.

ALL-IDB2 (Dataset1: 2-class): Accuracy calculations for ten self-trained and pre-trained models for dataset1 are shown in *table 2* for each model 5 runs are executed, and the average accuracy is noted. For self-training, ResNeXt-50 is showing the highest average accuracy of 99.33%, and for pretraining ResNet 101 is showing the highest average accuracy of 99.33%. From *table 2* it is observed that the self-trained model gave comparable accuracies to its pretrained counterparts. In the case of complex models belonging to the VGG family, it is observed that only self-trained models could achieve better average accuracy. The accuracy values for self-trained models of VGG11, VGG13 and VGG16 are 84.67%, 78.99%, 76.33%, respectively. Whereas the average accuracies for the pretrained counterparts in all three cases *i.e.*, VGG11, VGG13, and VGG16 gave the lowest accuracies of 41.67%, 51.99%, and 53.67% respectively.

CNMC (Dataset 2: 2-class): Accuracy calculations for ten self-trained and pre-trained models for dataset2 are shown in Table 3 for each model 5 runs are executed, and the average accuracy is noted. For self-trained models, ResNet-101 is showing the highest average accuracy of 91.25%, and for pre-

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 3 | Pages 694-704 | e-ISSN: 2347-470X

trained models, MobileNet is showing the highest average accuracy of 94.20%. From Table 3 it is observed that the self-trained model gave comparable results to its pretrained counterparts in Dataset 2 as well. In the case of complex models like those belonging to the VGG family same trend is observed. Only self-trained models could achieve better average accuracy of 89.29%, 67.87%, and 60.59% in the case of VGG11, VGG13, and VGG16, respectively. Whereas the average accuracies for the pre-trained counterparts in all three cases of the VGG family gave the lowest accuracies of 24.18%, 30.06 and 24.21% respectively. It is observed from *table 2 and table 3* that in case of both Dataset1 and Dataset2, ResNet-18, ResNet-34, ResNet-50, ResNet-101, WideResnet-50, ResNeXt-50, MobileNetV2 give almost comparable accuracies.

These results are surprising and call into question the understanding of the effects of ImageNet pretraining. Observations from the literature review suggest that ImageNet pretraining is a go-to solution (and probably will be for some time) when sufficient target data or computational resources are not available with the medical research community to make target task training possible. Both Dataset1 and Dataset2 are considered in the category of small datasets, but experiments and results have shown the effectiveness of Selftrained models. In the future, when the research community will have an ample amount of data and a faster calculation mode, the study undertaken in this work advocates the fact that training on target task is a solution to consider, especially when there is a significant gap between the source pretraining task and the target task. It is observed that the accuracies between pre-learning and self-learning are comparable for different models. In the case of VGG13, it is analyzed that the initialized random self-learning weights could converge, but this was not the case with the pre-trained models, which relied on the "best weights".

It is effortless to get accuracy above 90% for skewed datasets. In all the experiments under study, even if datasets were small, they were balanced, hence the accuracy values obtained are actual. As far as SelfTrained VGG11 and VGG13 results for Dataset2 are concerned, it is observed that VGG11 smoothened its outputs and converged through the entire epoch range. In contrast, VGG13 fluctuated a lot while finding suitable weights but eventually fell into random guessing in the forward epoch experiments. In pretraining, neither the VGG11 nor VGG13 model show convergence.

5.2 Discussion

Significant advantages of training from scratch (self-training) are depicted by the experimental results of this work. The experiments using self-trained models delivered comparable results in the setup of low data availability and also with weak data augmentation. It is also observed that self-training delivered appreciable classification accuracy when pretraining failed and when pretraining succeeded.

The main observations from the experiments are summarized as follows:

- Deep ImageNet models without architectural changes, are utilized for training from scratch on target tasks.
- Less number of iterations are required to converge in smaller models.
- The time required for ImageNet pre-training and self-training is almost similar in several experiments.
- Bigger, more complex models and pretrained models found it difficult to converge on the medical dataset.
- Self-trained and smaller, fewer complex models found it easier to converge on the medical datasets.

Here, the graphical representation of performance measures for dataset 2 is presented and discussed in *figures 4-8*. The performance of both ResNet 18 Selftrained and ResNet 18 Pretrained models is shown in *figure 4*. It shows the existence of overfitting after the epoch value of 150. The reason is, that training accuracy is observed to be better than test accuracy. However, in both cases, the accuracy follows an increasing trend after the epoch of 150. Training accuracy reaches the value of 100%, whereas test accuracy settles at around 85%.

In the case of the ResNet 50 self-trained and ResNet 50 pre-trained model, a different trend is seen. As observed from *figure 5*, overfitting is reduced, as test accuracy remains high as compared to training accuracy. In Selftrained and Pretrained ResNet50, till the epoch value of 120 and 150 respectively, the test accuracy remains high as compared to train accuracy. However, after that, there is a sudden fall, and the model fails to achieve accuracy greater than 50% in above mentioned two models.

Figure 6 depicts the performance measure i.e., accuracy for the self-trained and pre-trained ResNet101 model. Here, it is noticed that for self-trained ResNet101 the test and training accuracy attain similar value throughout the range of epochs and reaches up to 100%. In the case of pretrained ResNet101, the training accuracy is almost constant at around 80% whereas test accuracy is fluctuating between 50% and 85%. However, it remains higher than train accuracy for the complete epoch range.

Performance parameters, test and train accuracies for more complex and bigger models like VGG11, VGG13, and VGG16 are presented in *figure 7* and *figure 8*. For self-trained VGG11, initially up to the first 50 epochs, the test accuracy manages to be at a higher value than train accuracy. After an epoch of 50, the model enters into overfitting, delivering the highest training accuracy of 100% for epochs above 150. The test accuracy is maintained at around 85% from the initial epoch up to the final epoch. Whereas pre-trained VGG11 failed to achieve more than 25% accuracy in both the training and testing phases. The graphical analysis of the above discussion is shown in *figure 7*.

The performance measures for Self-trained VGG13 and VGG16 are depicted in the top two figures of *figure 8*. From this, it is observed that overfitting is reduced with test accuracy remaining higher than training accuracy. However, this effect is observed only up to epochs of 90 and 45 respectively in above mentioned two figures. From Figure 8 bottom two figures showing accuracy matrices for pre-trained

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 3 | Pages 694-704 | e-ISSN: 2347-470X

VGG13 and VGG16, it is analyzed that both the models are underperforming as far as smaller datasets are concerned.

A comparative analysis of the proposed work with the pretrained results obtained from the literature [47] for Dataset1 is presented in *table 4*. Here, the train test split is considered in the ratio of 60:40. As observed from *table 4*, a complex model like VGG16 is showing good performance i.e. 76.33% accuracy in the SelfTrained model as compared to respective pre-trained models in literature as well as in proposed work. The above discussion complements to main observation points discussed in *section 5.2*.

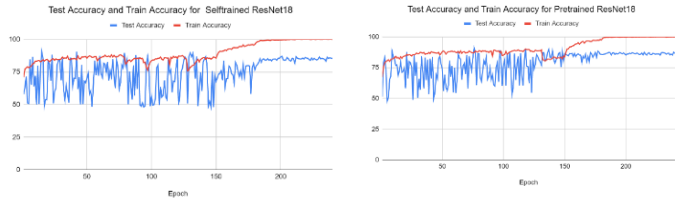

Figure 4: Test and train accuracy for self-trained and pre-trained ResNet18

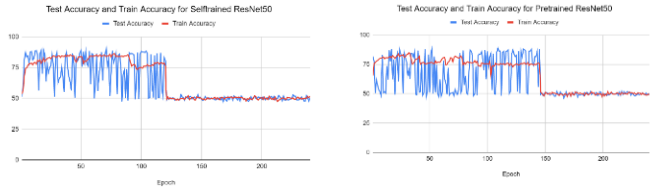

Figure 5: Test and train accuracy for self-trained and pre-trained ResNet50

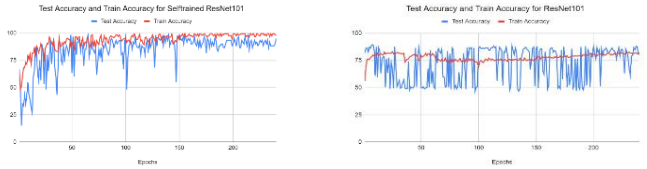

Figure 6: Test and train accuracy for self-trained and pre-trained ResNet101

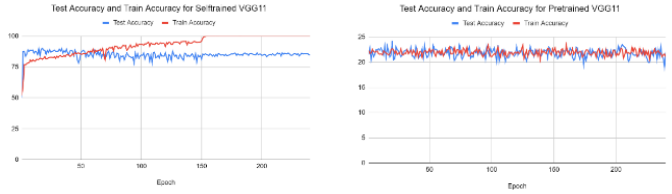

Figure 7: Test and train accuracy for self-trained and pre-trained VGG11

5.3 Limitations

- Testing on more datasets of blood smear images is required.
- Models like DenseNet, InceptionV2, and AlexNet are also very popular in image recognition and classification.
- Although, SGD was chosen for the very fact that it can't adapt. Ideally, a model should converge regardless of the type of optimizers used. So, SGD was able to show extreme conditions. But experimentation should be done with other optimizers as well.
- For performance measures, the F-score and other parameters of the confusion matrix are not explored.

6. CONCLUSION

With the results on different small datasets, it is observed that the training time is comparable between all the models and datasets. Looking at the results and training path, it can be concluded that the pretraining of models for small medical datasets is not recommended. Excellent accurate accuracy for the problem at hand can be achieved provided a suitable, non-complex model that can understand and learn all the features of the given dataset is chosen. In this work, self-training a model from scratch with an application aiming for image classification is studied. Obtained experimental results highlight that self-training works better, even if the amount of data available is less. So, such models can also be explored in the field of speech recognition, Natural Language Processing (NLP), bioinformatics, etc. In this work, publicly available data sets are studied which belong to a particular type of blood disease, i.e., Leukemia. The positive effect of this method is expected to give good results if applied to datasets in the field of self-driving and healthcare. Also, the method can be adapted for other datasets and sensitive applications that have ethical implications, such as mass surveillance.

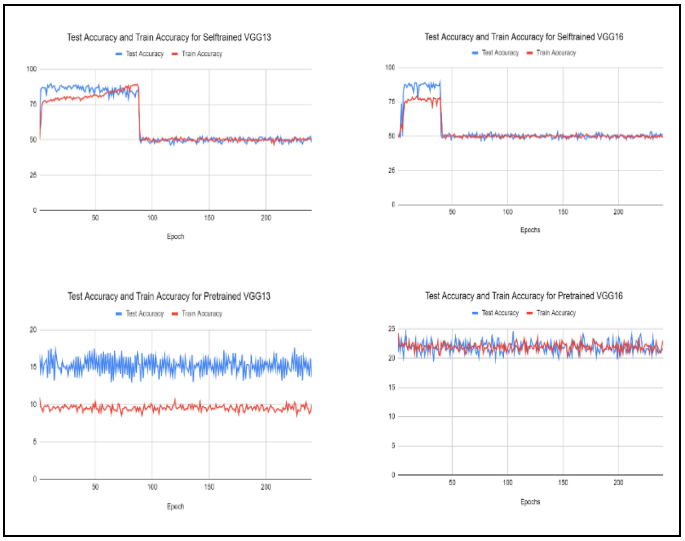

Figure 8: Test and train accuracy for self-trained and pre-trained VGG13 and VGG16

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 3 | Pages 694-704 | e-ISSN: 2347-470X

Table 1. Parameter setting of CNN

Parameters	CNN Architecture									
	Resnet-18	*Resnet-34*	*Resnet-50*	*Resnet-101*	*VGG11*	*VGG13*	*VGG16*	*WideResnet50*	*ResNext50*	*MobilenetV2*
Optimizer	SGD	SGD	SGD	SGD	SGD	SGD	SGD	SGD	SGD	SGD
Learning Rate	0.05	0.05	0.05	0.05	0.05, 0.01	0.05, 0.01	0.05, 0.01	0.05	0.05	0.01
Learning Decay Rate	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
Momentum	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
Epochs	240	240	240	240	240	240	240	240	240	240
Train Batch size	8	8	8	8	8	8	8	8	8	8
Test Batch Size	8	8	8	8	8	8	8	8	8	8
Total No of Parameters	11M	21,8M	25.5M	44.5M	138M	138M	138M	2.2M	25.0M	4.3M

Table 2. Accuracy of Pretrained and Selftrained Model-Dataset1

Models	Self-Training time per epoch(sec)	Pretraining time per epoch(sec)	Self-Training Average Accuracy (%)	Pretraining Average Accuracy (%)
MobileNet	2	1.9-2	98.66	98.33
ResNet18	1.5	1.4-1.5	98	98.32
Resnet34	1.9	1.9	97.768	98.666
Resnet50	2.7	2.6-2.7	95.19	95.99
Resnet101	5.1	5	96.67	99.33
Resnext50	4	3.6	99.33	90.67
VGG11	2.7	2	84.67	41.67
VGG 13	3.3	3.2	78.99	51.99
VGG16	3.7	3.6	76.33	53.67
WiderResnet50	4.3	4.2-4.3	97	98

Table 3. Accuracy of Pretrained and Selftrained Model-Dataset2

Model	Self-Training time per epoch(sec)	Pretraining time per epoch(sec)	Self-training Average Accuracy (%)	Pretraining Average Accuracy (%)
MobileNet	38	36	90.76	94.2
ResNet18	26	30	89.81	90.22
ResNet34	36	43	90.24	90.05
ResNet50	62	55	90.23	89.98
ResNet101	76	74	91.25	89.41
ResNext50	67	66	90.27	89.72
VGG11	47	47	89.29	24.18
VGG 13	30-31	58	67.87	30.06
VGG16	67-68	67	60.59	24.21
WideResnet50	80	78	90.28	81.45

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 3 | Pages 694-704 | e-ISSN: 2347-470X

TABLE 4. Performance comparison of the proposed work with the pre-trained model in the literature

DEEP MODELS	ACCURACY		
	FROM LITERATURE	PROPOSED WORK	
	PRE-TRAINED	SELF-TRAINED	PRE-TRAINED
DATASET1			
VGG16	71.35	**76.33**	53.67
MOBILENETV2	92.89	**98.66**	98.33
RESNET 18	89.23	**98**	98.32

REFERENCES

[1] B. J. Bain, "Diagnosis from the blood smear," New England Journal of Medicine, vol. 353, no. 5, pp. 498-507, 2005.

[2] P. G. Gallagher. Red cell membrane disorders. ASH Education Program Book, vol. 1, pp. 13-18.

[3] T. J.Durant, E.M.Olson, W.L.Schulz, R.Torres, "Very deep convolutional neural networks for morphologic classification of erythrocytes," Clinical Chemistry, vol. 63, no.12, 2017, pp. 1847-1855.

[4] J. Ford, "Red blood cell morphology," International journal of laboratory hematology, vol. 35, no. 3, pp. 351-357, 2013.

[5] V.Sanjay and P.Swarnalatha, "A survey on various machine learning techniques for an efficient brain tumor detection from MRI images," IJEER, vol. 10, no. 2, pp. 177-182,2022.

[6] H.Singh and R.Singh Solanki, "Classification and feature extraction of brain tumor from MRI images using modified ANN approach," IJEER, vol. 2, no. 2, pp. 10-15,2021.

[7] J. Zhao, M. Zhang, Z. Zhou, J. Chu, "Automatic detection and classification of leukocytes using convolutional neural networks," Medical & biological engineering & computing, vol. 55, no. 8, 2017, pp. 1287-1301.

[8] G. Litjens et al.Deep learning as a tool for increased accuracy and efficiency of histopathological diagnosis. Scientific reports, vol. 6, no. 1, 2016 pp. 1-11.

[9] C. McClanahan, "History and evolution of GPU architecture," A Survey Paper, vol. 9, 2010.

[10] Ciresan. D., Meier. U., Masci, J., Gambardella.L., and Schmidhuber, J., 2011. Flexible, high-performance convolutional neural networks for image classification. Twenty-second international joint conference on artificial intelligence.

[11] Deng, J., Dong,W., Socher,R., Li,Li-Jia., Li, K., Fei-Fei, Li., 2009. Imagenet: A large-scale hierarchical image database.IEEE conference on computer vision and pattern recognition.

[12] O. Russakovsky et al., "Imagenet large scale visual recognition challenge," International journal of computer vision, vol. 115, no. 3, pp. 211-252, 2015.

[13] A. Krizhevsky, I. Sutskever, G. Hinton, "Imagenet classification with deep convolutional neural networks," Advances in neural information processing systems, vol. 25, 2012.

[14] Zeiler, M. and Fergus, R. 2014.Visualizing and understanding convolutional networks. European conference on computer vision, Springer, Cham.

[15] A. G. Howard, et al, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," 2017.

[16] F. Iandola, M. Moskewicz, S. Karayev, R. Girshick, T. Darrell and K. Keutzer, "Densenet: Implementing efficient convent descriptor pyramids," 2014.

[17] M. Sajjad, et al., "Leukocytes classification and segmentation in microscopic blood smear: a resource-aware healthcare service in smart cities," IEEE Access, vol. 5, pp. 3475-3489, 2016.

[18] J. Prinyakupt and C. Pluempitiwiriyawej, "Segmentation of white blood cells and comparison of cell morphology by linear and naïve Bayes classifiers," Biomedical engineering online, vol. 14, no. 1, pp. 1-19, 2015.

[19] A. Abdeldaim, A. Sahlol, M. Elhosney, A.Hassanien, "Computer-aided acute lymphoblastic leukemia diagnosis system based on image analysis," Advances in Soft Computing and Machine Learning in Image Processing, Springer, Cham, 2018, pp. 131-147.

[20] Vogado, L., Veras, R., Andrade, A., Araujo, F., Silva, R. and Aires, K. 2017.Diagnosing leukemia in blood smear images using an ensemble of classifiers and pre-trained convolutional neural networks.30th SIBGRAPI Conference on Graphics, Patterns, and Images (SIBGRAPI). IEEE.

[21] S. Pan, and Q. Yang, "A survey on transfer learning." IEEE Transactions on knowledge and data engineering," vol. 22, no.10, pp.1345-1359, 2009.

[22] Torrey, L. and Shavlik, J. 2010. Handbook of research on machine learning applications and trends: algorithms, methods, and techniques. IGI Global.

[23] Oquab, M., Bottou, L., Laptev, I. and . Sivic, J. 2014. Learning and transferring mid-level image representations using convolutional neural networks. Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1717-1724.

[24] You, Q., Luo, J., Jin, H., and Yang, J. 2015. Robust image sentiment analysis using progressively trained and domain transferred deep networks. Twenty-ninth AAAI conference on artificial intelligence.

[25] Yao, Y. and Doretto, G. 2010. Boosting for transfer learning with multiple sources. 2010 IEEE computer society conference on computer vision and pattern recognition. IEEE.

[26] N. Srivastava, G. Hinton, A. Krizhevsky, "Dropout: a simple way to prevent neural networks from overfitting," The journal of machine learning research, vol. 15,no. 1, pp. 1929-1958, 2014.

[27] Xie,S., Yang, T., Wang, X. and Lin, Y. 2015 . Hyper-class augmented and regularized deep learning for fine-grained image classification. Proceedings of the IEEE conference on computer vision and pattern recognition.

[28] Wong, S., Gatt, A., Stamatescu, V. and Mcdonnell, M. 2016. Understanding data augmentation for classification: when to warp? 2016 international conference on digital image computing: techniques and applications (DICTA). IEEE.

[29] L. Perez, and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," 2017.

[30] T. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, Y. Ma, "PCANet: A simple deep learning baseline for image classification?," IEEE transactions on image processing, vol. 24, no. 12,pp. 5017-5032, 2015.

[31] H. C. Shin, et al., "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics, and transfer learning." IEEE transactions on medical imaging, vol. 35, no. 5, pp. 1285-1298, 2016.

[32] B. Q. Huynh, H. Li, and M. L. Giger, "Digital mammographic tumor classification using transfer learning from deep convolutional neural networks," Journal of Medical Imaging, vol. 3, 2016.

[33] M.Geng, Y. Wang, T.Xiang, Y.Tian, "Deep transfer learning for person re-identification," arXiv preprint arXiv:1611.05244, 2016.

[34] A. Kumar, J. Kim, D. Lyndon, M. Fulham, and D. Feng, "An ensemble of fine-tuned convolutional neural networks for medical image classification," IEEE journal of biomedical and health informatics, vol. 21, no.1, pp. 31-40, 2016.

[35] Yanai, K. and Kawano, Y. 2015. Food image recognition using a deep convolutional network with pre-training and fine-tuning. 2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW). IEEE.

[36] Mersha Nigus and H.L Shashirekha (2022), A Comparison of Machine Learning and Deep Learning Models for Predicting Household Food Security Status. IJEER 10(2), 308-311. DOI: 10.37391/IJEER.100241.

[37] Carneiro, G. and Vasconcelos, N. 2005. Formulating semantic image annotation as a supervised learning problem. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). vol. 2. IEEE.

[38] Fergus,R., Perona, P. and Zisserman, A. 2003.Object class recognition by unsupervised scale-invariant learning. 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Proceedings, vol. 2. IEEE.

[39] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wo, A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," 2011.

[40] Liu, S. and Deng, W. 2015. Very deep convolutional neural network-based image classification using a small training sample size.2015 3rd IAPR Asian conference on pattern recognition (ACPR), IEEE.

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 3 | Pages 694-704 | e-ISSN: 2347-470X

[41] S. Hira, Anita Bai, and S. Hira, "An automatic approach based on CNN architecture to detect Covid-19 disease from chest X-ray images," Applied Intelligence, vol. 51, no.5, pp. 2864-2889, 2021.

[42] M. Loey, M. Naman, and Z. Hala, "Deep transfer learning in diagnosing leukemia in blood cells," Computers, vol. 9, no. 2, 2020.

[43] Habibzadeh, M., Jannesari, M., Rezaei,Z., Baharvand, H. and Totonchi, M. 2017. Automatic white blood cell classification using pre-trained deep learning models: ResNet and Inception. Tenth international conference on machine vision (ICMV 2017), vol. 10696, International Society for Optics and Photonics.

[44] V.Narasimha and Dr.M.Dhanalakshmi, "Detection and severity identification of Covid-19 in chest X-ray images using deep learning," IJEER, vol. 10, No. 2, pp. 364-369.

[45] Labati, R., Piuri, V. and Scotti, F. 2011.All-IDB: The acute lymphoblastic leukemia image database for image processing. 2011 18th IEEE international conference on image processing. IEEE.

[46] Duggal, R., Gupta, A., Gupta, R., Wadhwa, M. and Ahuja, C. 2016. Overlapping cell nuclei segmentation in microscopic images using deep belief networks. Proceedings of the Tenth Indian Conference on Computer Vision, Graphics and Image Processing.

[47] Das, P. K., and S. Meher. 2021. An efficient deep convolutional neural network-based detection and classification of acute lymphoblastic leukemia. Expert Systems with Applications 183:115311. doi:10.1016/j.eswa.2021.115311.