Journal of Theoretical and Applied Information Technology
30th November 2021. Vol.99. No 22

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

HYPERPARAMETER OPTIMIZATION IN CUSTOMIZED CONVOLUTIONAL NEURAL NETWORK FOR BLOOD CELLS CLASSIFICATION

GHAZALA HCINI[1] , IMEN JDEY[2],ASHRAF HENI[3] , HELA LTIFI[4]

[1]Faculty of Sciences and Technology of Sidi Bouzid, University of Kairouan, Kairouan, Tunisia

E-mail: [1]hcinighazala@fstsbz.u-kairouan.tn, [2]imen.jdey@fstsbz.u-kairouan.tn, [3] Ashraf.hani@fstsbz.u-kairouan.tn, [4] hela.ltifi@fstsbz.u-kairouan.tn

ABSTRACT

Recently, various uses of supervised classification recognition algorithms for medical images are reported in literature. Specifically, in the current deep learning era, machine learning techniques are considered as the most important and used approach for automatic healthcare systems. In this context, many comparisons of supervised deep learning techniques, more precisely, the neural one, are proposed. The proposed approach provides a medical assistance based on relevant aspects of Machine -Learning methods applied for blood cells objects recognition while taking into consideration the property of uncertainty of this kind of image. The overview presented in this article examines the existing literature and the contributions already done in the field of intelligent healthcare systems for blood cell images classification. For this purpose, we summarize previous efforts made to define recognition process with supervised deep learning method, establishing a novel definition of personalized Machine- Learning with a major focus on the uncertainty input image. Departing from this definition, we propose and discuss the efficiency of Convolutional Neural Network for which the architecture is built and examined in detail. A Bayesian optimization of Convolutional Neural Network hyper parameters is also proposed. The main goal is to increase recognition rate while respecting time complexity. That is why an experimental comparison of Convolutional Neural Network with Support Vector Machine and K- nearest neighbor performance is discussed.

Keywords: *Blood Cell Images, Machine Learning, Deep Learning, Convolutional Neural Network, Bayesian optimization*

1. INTRODUCTION

Artificial intelligence is a vast computing field that allows machines to learn and gain knowledge from experience, adapt to data, and do human-like jobs intelligently, but without awareness, sensibility or mind. [1] [2] [3]. Deep learning, a subfield of it, contributes to the high quality of health-care services [4]. Its techniques are effective tools for augmenting classical machine learning and allowing computers to learn from data in order to create intelligent applications [5]. Because deep learning is achieving amazing results even at human level performance [6], it is currently accelerating the lively development of artificial intelligence in various fields and has had profound impacts on scientific fields such as computer vision and natural language processing [7] as well as medical image processing [8] [9] [10] [11] [12], even when the data set is unstructured and diverse [13]. We focus on intelligent health care systems in this paper by classifying blood cell images. Classifying cell types using statistical and moment invariant analysis is used to assess clinical state in order to entail the classification and identification of a blood sample from patients. In fact, we recommend that Convolutional Neural Network (CNN) algorithms be customized to deal with this specific image type, which is characterized by uncertainty and imprecision during the acquisition process. In order to provide context, we begin with a background study, followed by a discussion of artificial intelligence and related fields in the second section. We also go through the convolutional neural network architecture in detail. The final portion is to determine the deep learning approaches used to recognize blood cells images. In section 4, the proposed Convolutional Neural Networks technique is described by specifying the architecture performing by Bayesian optimization of convolutional neural network hyper parameters in order to improve recognition rate. Then, in Section 5, the experimental study. In section 6, an experimental comparison with similar methods like as support vector machine and K- nearest neighbor is discussed in the assessment optics of our proposed system. Section 7 is where the conclusions are drawn.

2. BACKGROUND OF THE STUDY

White blood cells are important components of the human immune system. Blood cells are divided into three types: red blood cells (RBCs), which transport oxygen, white blood cells (WBCs), which are the face of the immune system, and platelets, which cause blood to clot in damaged tissues. One of the hurdles in medical science is identifying and quantifying the number of white blood cells. The quantity of red blood cells is the main reason. The WBCs account for about 1% of the total blood volume in a healthy adult. Because of the low proportion of WBCs in blood, identifying WBCs is difficult, making the task of defining WBC subtypes even more difficult [14]. The classification of White Blood Cells is critical in the detection of sickness in a person. Infections, allergies, anemia, leukemia, cancer, Acquired Immune Deficiency Syndrome (AIDS), and other disorders can all benefit from the classification [14]. Also the identification of red blood cells (RBCs) is a critical step in the early detection of blood-borne diseases including malaria and anemia, which must be followed by appropriate treatment [15].

To get around the manual gating limitations which is a technique, applied for the classification of blood cells images, that is expert-driven and introduces two major sources of operator bias. For starters, scatter plot gates are very subjective, and so might change significantly amongst operators. Automating the gating process with image processing technology combined with artificial intelligence and computer graphics systems to provide approaches that are more accurate, faster, and simple to use. Machine learning (ML) and Deep Learning (DL) could be feasible approaches to address these disadvantages which is the difficulty to duplicate, and that it takes a long time for large experiments [16]. The goal of machine learning, a part of artificial intelligence, is to combine the data we see with the experience that the computer gains in order to generate information that we can use. Supervised and unsupervised learning are the two broad categories of machine learning that can be used to solve problems involving classification and regression. Support Vector Machine (SVM), one of the most extensively used machine learning algorithms before neural networks became mature, is one of its methodologies [17] [18] [19] [20] [21]. And K- nearest neighbor (KNN). This technique is one of the most fundamental categorization and recognition techniques. That is why, in the last section, we employ it in a comparative study with our proposed approach to evaluating performance.

2.1 Support Vector Machine (SVM)

Before neural networks develop, the Support Vector Machine method is one of the most popular supervised learning algorithms [8]. The goal of this method is to identify the best separation between two classes of objects. Indeed, in order to optimize the distance between the hyperplane and the observations closest to the training sample, it chooses the hyperplane that divides the set of observations into two distinct classes (figure 1 and figure 2). Most of the time, the complexity of this strategy is polynomial. Indeed, it is mostly determined by the kernel function utilized [22], such as Gaussian, polynomial [23], and others, as inspired by [24] [25] [26].

Kernel Function:

$$K(x_i, x_j) = \emptyset(x_i)^T (x_j) \quad (1)$$

$$f(x_i) = \sum_{n=1}^{N} (\alpha_n \gamma_n K(x_n, x_i) + b \quad (2)$$

Where, x_n represent support vector data, α_n is Lagrange multiplier and γ_n is the label of membership class (+1, -1) with n=1, 2, 3, 4,...., N. [27]

- **Polynomial**:

$$K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0 \quad (3)$$

- **Gaussian**:

$$K(x_i, x_j) = \exp\left(-\frac{\|xi - xj\|^2}{2\gamma^2}\right), \gamma > 0 \quad (4)$$

- **Linear**:

$$K(x_i, x_j) = x_i^T x_i \quad (5)$$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

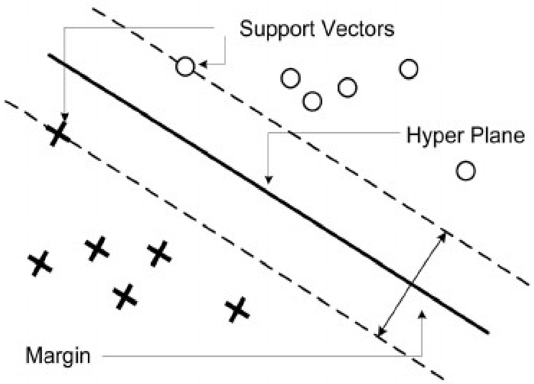

Figure 1: Support Vector Machine(SVM) for linear data **[28]**

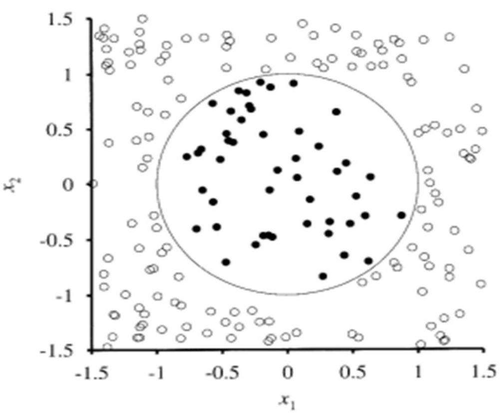

Figure 2: Support Vector Machine(SVM) for non-linear data **[29]**

2.2 K-Nearest Neighbor (KNN)

The k-nearest neighbor algorithm is a method for categorizing objects (figure 3). It is relatively simple classification model [30]. For this approach, the training procedure is limited to storing the vectors of different features as well as the labels of the training images. The unlabeled question point is simply assigned to the label of its nearest k neighbors throughout the classification procedure [3]. This is one of the most fundamental categorization and recognition methods [7].

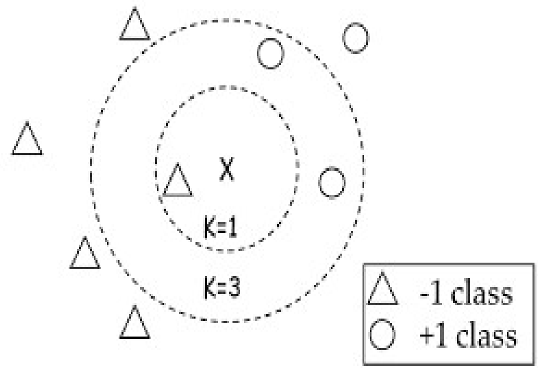

Figure 3: Illustration example of KNN algorithm **[31]**

What is the best value for K?

In actuality, there is no one-size-fits-all approach for selecting K; instead, it is dependent on the task at hand. If K is too large, the algorithm may incorrectly classify the new point because its nearest neighbors are located far away. Because of the noise in the training data, if k is too little, the algorithm is prone to overfitting the data. This will have an impact on the ability to generalize. The K size that minimizes the classification error is the best [32].

Distance function

The technique works by calculating the distance between the mathematical values of these points. It finds the chance of the points being similar to the test data by computing the distance between each data point and the test data. The distance function can be Euclidean. The shortest distance between two points, regardless of their dimensions, is known as the Euclidean distance. The Euclidean distance is the most used method for calculating the distance between two points.

$$\mathrm{dist}((x, y), (a, b)) = \sqrt{(x - a)^2 + (y - b)^2} \quad (6)$$

To distinguish the difference between the two machine learning methods mentioned above Support Vector Machine (SVM) and k-nearest neighbor (KNN), the table below shows its pros and cons (table1).

Table 1: Comparison table on the most classifiers used in machine learning Support Vector Machine(SVM) and k-nearest neighbor(KNN) its pros and cons **[33]**.

Algorithm	Advantages	Disadvantages
SVM	High accuracy, even when data is not linearly separable.	- High memory requirements - High level of complexity

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

KNN	It can be robust in the face of noisy training data at times.	It makes no inferences from the training data and instead relies solely on the training data for classification.

2.3 Deep Learning

With the advancement of machine learning and the exponential growth of data [34], scientific research on Artificial Intelligence has shown significant limitations [35] associated with various machine learning techniques. This inspired data scientists to create the notion of deep learning [36] [37], a new machine learning paradigm [38] [39] based on artificial neural networks that consumes millions of data points [40] [41]. It has surpassed Machine-Learning algorithms in popularity (figure4) [42] [43].

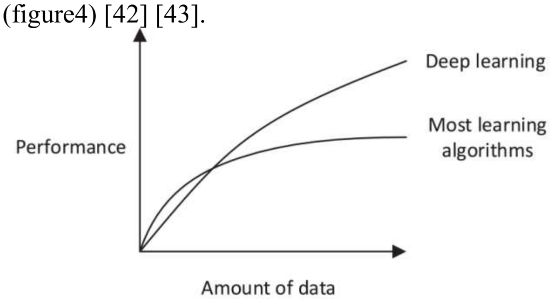

Figure 4: Consistently increasing data provides better results **[44]**

Deep learning can be used to solve data analysis problems and produce high-quality summaries in order to make better decisions [45]. It has an impact on a variety of fields, with computer vision [46] [38] being one of the most notable. Object detection [38] [47] [48], face recognition [49], and other tasks like activity or event recognition [50] are examples of computer vision challenges. We are interested in the basic visual recognition application of object detection, which has had a lot of success in computer vision [51] [52]. When it comes to complicated problems like image classification, we find that deep learning really shines [53] [10] [54].

Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) are two common deep learning architectures for supervised learning [55] [56]. The hidden state of a recurrent neural network is one of its most notable features, as it aids it in remembering sequence information . Gated Recurrent Unit (GRU) and Long Short Term Memory (LSTM) are two common types of recurrent neural networks (LSTM). RNN saves all of the data that was previously computed. It employs the same parameter for all input and hidden layers, which lowers parameter complexity [57]. RNNs can handle time series and are utilized in a variety of applications such as stock price prediction, voice recognition, and video processing. The ability of this sort of network to preserve observations in order to extract information distinguishes it from others [56]. In the realm of computer vision, CNN is the most often utilized [58] [9]. It has had a good track record in practical applications [59]. Convolutional neural networks are widely used image classification models [60] [41] [61] [43] [62], and they have been found to perform well in this field [63] [58] [64] [65] [66]. The key advantage of CNN is that it has spread due to three factors; (1) it eliminates the need for manual feature extraction [9] because the key learns them automatically, (2) it is capable of being retrained to do new recognition tasks and gives excellent recognition results [67], and (3) you can also build on top of existing networks [68].

A CNN is a common deep learning architecture that learns relevant feature representations from image data [69] automatically [70]. It is not easy to come up with a powerful model to extract these attributes. As a result, we will need to go through the CNN architectural components in the next part.

2.4 Convolutional Neural Network Architecture

Convolutional Neural Networks get their name from the sort of hidden layers that make them up. Image and video recognition, recommendation systems, and natural language processing are just a few of the areas where they can be used [71]. CNN has produced excellent results in real applications [72]. The name Convolutional Neural Network declares that the network employs conversion as a mathematical process [53]. Convolutional Neural Networks [73] [9] [41] [74] have a variety of layers (figure 5).

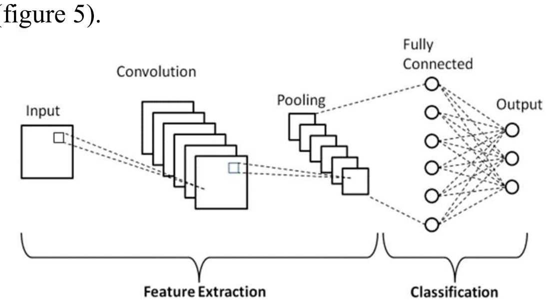

Figure 5: A Convolutional Neural Network architecture: each of the components to be modeled is represented by a layer inspired from **[75]**

2.4.1 Convolution layer

This layer represents the key component of Convolutional Neural Networks [9]. The main purpose of this layer is to identify the existence of a set of features in the images received as input [76]. Filtering is carried out by convolution: its principle is to slide a window showing the feature on the image, and a product is calculated between the feature and each fragment of the image purged.

2.4.2 Pooling layer

This layer is generally installed between two convolution layers. The pooling operation is used to reduce the size of the images received as input, with all protection of important features. The processing is done by cutting the image into regular cells while keeping the maximum value of each cell.

2.4.3 Rectified Linear Unit (ReLU) layer

This layer acts as an activation function. It is a non-linear operation which replaces all negative values received with Zeros. The purpose of this operation is to introduce non- linearity in the model.

$$ReLU(x) = \max(0,x) \quad (7)$$

2.4.4 Fully-connected layer

It is the last layer of the Convolutional Neural Network. Its role is the input image classification. The fully-connected layer takes all neurons in the previous layer (Convolutional layer or pooling layer) and connects it to every single neuron it has.

3. RELATED WORKS

Researchers about blood cell image classification in recent years are mainly based on Deep learning techniques.

Shafique, S., & Tehsin, S. [77] used pretrained model AlexNet to automate the detection of acute lymphoblastic leukemia and the classification of its subtypes into four classes, applied on a public dataset which was split into two parts. Acute lymphoblastic leukemia IDB 1 was composed of 108 images, containing 59 images from healthy individuals and 49 images from leukemia patients. They used a data augmentation approach called image manipulation, in which they conducted image rotation and mirroring to improve the training data.

G. Liang & all [78] proposed an architecture that combined CNN and RNN in order to propose the CNN-RNN framework that consists of two different sections: the CNN section, which uses the Xception model, and uses the transfer learning method; the other section is the RNN, which uses the Bi-directional LSTM model applied to process a data set retrieved from the BCCD dataset (https://github.com/Shenggan/BCCDDataset), and publicly available dataset (https://www.kaggle.com/paultimothymooney/bloodcells/data). It contains 12444 augmented images of blood cells (JPEG), with 9957 and 2487 training and test images, and four additional subtype Eosinophil (2497 images), lymphocyte (2483 images), Monocyte (2478 images), and Neutrophil (2499 images). The entire network used RGB images of size 320*240*3 pixels.

Yildirim, M., Çinar, A. [79] proposed a model CNN that was coupled with AlexNet, Resnet50, Densenet201 and GoogleNet applied to Kaggle Dataset which consists of 4 different leukocyte cells (Eosinophil, Lymphocyte, Monocyte and Neutrophil) using Gaussian and median filters.

Andrea Acevedo & all [80] proposed a personalized CNN trained to classify eight classes of cells (Neutrophils, Eosinophils, Basophils, Lymphocytes, Monocytes, im-mature granulocytes (myelocytes, metamyelocytes and promyelocytes), Erythroblasts and Platelets) using a data set over 17000 cells images (88% Train, 12% Test) (RGB, 360*363 pixels) obtained from clinical practice. A confusion matrix was calculated to obtain performance parameters such as sensitivity, specificity and precision.

In [81], Sahlol & all improved Swarm optimization of deep features for efficient of white blood cells provided by Department of Information Technology. The input image is (224, 224, 3). They extracted the features using the VGGNet, then filtered these features using statistically enhanced Salp Swarm Algorithm (SESSA). This algorithm selected only 1K out of 25K features.

In [82], Loganathan, V. used the transfer learning approach to move the pre-trained weight parameters from the Image Net dataset to the CNN portion, and a custom loss function to make our network train and converge faster and with more accurate weight parameters. As a result, using segmentation and classification, this procedure is used to discover the irregularity of the cell. They ran numerous tests to assess the performance of the deep CNN in the special RBC classification scenarios, and they compared the findings. They used 434 raw microscopy images of 8 distinct SCD patients obtained from two separate hospitals in their trials to validate the resilience of their methods in dealing with varied imaging data. The classification results showed that KNN had an accuracy of 88.04 % while ANN had an accuracy of 54 %.

Aliyu, Hajara Abdelkarim, & al. [83] used a framework that included image acquisition (were taken from 130 patients at Murtala Muhammad Specialist Hospital (MMSH) and Aminu Kano Teaching Hospital (AKTH), both in Nigeria, with authorized ethics numbers of MOH/off/797/T.1/849 and NHREC/21/08/2008/AKTH/EC/2307), image pre-processing (for further processing, it is necessary to remove undesired noise), automatic RBC blotch extraction (which was based on parameters like eccentricity, area, and bounding box), and the AlexNet model that had been trained.

Abou El-Seoud, S., Siala, M., & McKee, G. [84] proposed an approach based on convolutional neural network to detect and classify normal white blood cells. There are five layers in the proposed CNN structure. The first three layers are used to extract features, while the latter two levels are used to classify the features. The input image was [50x50x3], and the size of the receptive field (also known as filter) was 5x5. In each iteration, the stride is 1 and the filter is moved one pixel. On a dataset of 10,000 blood cell images, the proposed model was used.

The table 9 (*Summary of Deep Learning based Blood Cells Diagnosis)* displays the results of these different deep learning methods based on blood cells diagnosis.

As we notice not only the techniques, with the major core architectures of deep learning models Convolutional Neural Network(CNN), Long Short-Term Memory (LSTM) [39] Xception, Inceptionv3 [85] and the different pretrained models like ResNet50, AlexNet [86], DenseNet201, GoogleNet and Vgg-16 [87] [86] [4] [88] currently used by the authors have an effect on the results, but also the number of images for the processing, which affirms that deep learning has played an important role in Big Data analytical solutions [89] [90].

4. PROPOSED METHOD: CONVNET

The aim is to identify and characterize blood samples from patients. Several methods of detecting and classifying blood cells subtypes have important medical applications. CNN is a type of Artificial Neural Network (ANN) [34] [9]. The use of CNN for deep learning has spread for the main reason: the production of perfect recognition results [63]. It takes raw pixels as input and gives a result corresponding to the probabilities that the inputs belong to different classes. In the deep learning workflow using CNN, there are several repeated steps the until network reaches the desired level of accuracy such as (1) preprocess images, (2) set training options, (3) train network, and (3) testing to deploy the model.

The architecture of our model contains three Convolutional layers with two pooling layers installed between the Convolutional layers. It can have different combinations of initial layers but usually end with fully connected layer, a Softmax layer or a classification layer.

Hyper parameters related to our network structure

Important factors that affect the training time and the value of accuracy are notably, Network architecture, mini Batch Size and initial learning rate, also the choice of optimizer. Several optimizers are used in artificial intelligence. The best known are Stochastic Gradient Descent with momentum SGDM and Adaptive Moment Estimation ADAM [91] [92] [93], to minimize the loss. We trained the model and evaluated these optimization techniques in terms of convergence speed, accuracy and loss function. Also experimentation using certain activation functions, among which the most used are ReLU and tanh [93].

CNN Schematic (figure 6)

- The first Convolutional layer consists of 8 number of 4*4 sized filters
- The second Convolutional layer has 16 number of 2*2 filters
- The third Convolutional layer has 32 number of 6*6 filters
- Two max-pooling layers each of size are also being used in the network.
- The padding type used is "same", the size of the input matrix after each convolution does not change.
- An average pooling layer; its role is to sub-sample by dividing the input into rectangular grouping regions and calculating the mean values for each region.
- A FullyConnectedLayer represented the output layer.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

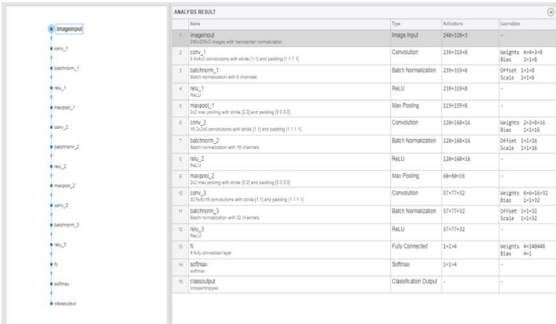

Figure 6: ConvNet is a Convolutional Neural Network that is 15 layers deep. It reports zero errors

5. EXPERIMENTAL STUDY

The used dataset for our study contains blood cells images (JPEG) with cells type labels. The cell types are Neutrophil, Monocyte, Lymphocyte, and Eosinophil (figure 7). The folder dataset used contains 12444 augmented images and 4 additional subtype labels (JPEG+CSV). The size of the images is [240*320], 3 channels (RGB). https://www.kaggle.com/paultimothymooney/blood cells. We want to automatically classify each image based on what type of cells it contains.

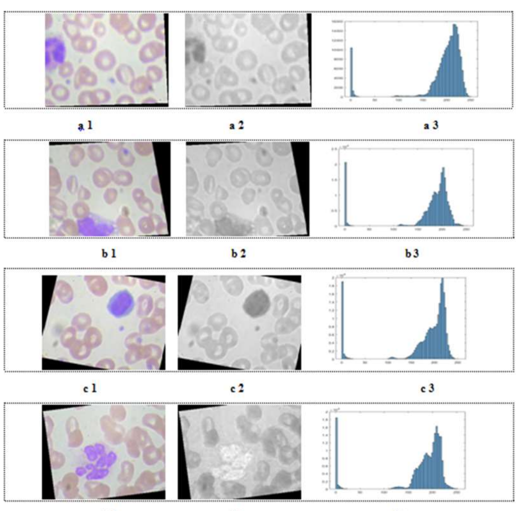

Figure 7: Cell types: Example images of 4 classes; (a1)Neutrophil image, (a2)grayscale Neutrophil image, (a3)Histogram, (b1)Monocyte image , (b2)grayscale Monocyte image, (b3) Histogram, (c1) Lymphocyte image, (c2)grayscale Monocyte image, (c3) Histogram, d1) Eosinophil image, (d2)grayscale Eosinophil image, (d3)Histogram. The classification is performed by automatically classifying the dataset of 4 classes.

To provide a quick overview, the following features and functions of WBC types and subtypes, as well as information on WBC are described [94]:

Neutrophils: are an important line of protection against germs and a part of the innate immune system.
Monocytes: activate osteoclasts, which are cells that can destroy bone. They are the most numerous WBCs.
Lymphocytes: are classified by their size and cytoplasmic granularity, and depending on their maturation stage, they might have a tiny or large nucleus.
Eosinophils: have the ability to release toxins from their granules, which can be used to destroy infections like parasites and worms. Their big granules make them easy to spot in stained smears. Eosinophil nuclei frequently have two lobes joined by a band of nuclear material.

5.1 CNN Architecture: ConvNet

This section provides the detailed description of the proposed architecture ConvNet using the blood cells dataset. Image preprocessing is the first stage, which includes image reading and image labeling. After the input images have been preprocessed. A convolutional neural network would be used to process the images (figure 8).

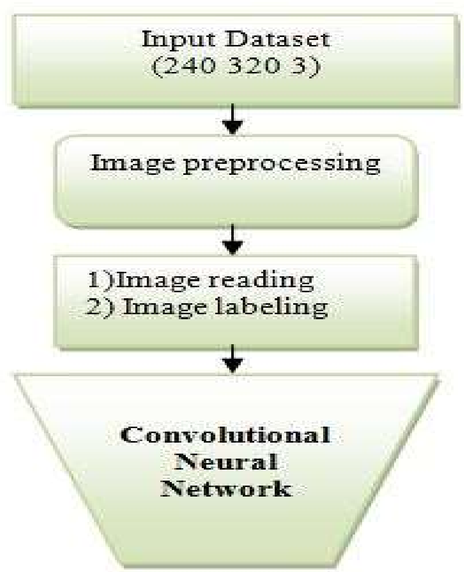

Figure 8: Proposed model stages

We separated the data into a training and test dataset. We analyzed the shapes of the training and test datasets . The training dataset was found to be consisting of 9957 images, 2497 Eosinophil, 2483 Lymphocyte, 2478 Monocyte and 2499 Neutrophil.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The test dataset had 2487 images, 623 Eosinophil, 620 Lymphocyte, 620 Monocyte and 624 Neutrophil.
A study was made about optimization technique. We trained our model ConvNet and evaluated the optimization techniques Stochastic Gradient Descent with momentum SGDM and Adaptive Moment Estimation ADAM in terms of the value of accuracy. On the practical side, the Gradient Descent consists of evaluating the gradient of the cost function, then updating the parameters (the weights of a neural network) to reduce the error. The table below shows the significant impact of SGDM compared to ADAM.

Validation criteria:

We used accuracy to evaluate the proposed method's performance. The definition of this measure is as follows:

$$\textbf{Accuracy} = \frac{\mathbf{TP+TN}}{\mathbf{TP+TN+FP+FN}} \quad (8)$$

Where "TP" (true positives) refers to malignant samples that the classifier correctly labels, and "TN" (true negatives) refers to benign samples that the classifier correctly labels. False positives (FP) are malignant cells that have been mislabeled as benign, while false negatives (FN) are benign samples that have been mislabeled as malignant.

Table 2: Classification accuracy for both SGDM and ADAM

Optimizer	Epochs	Max. iteration	Accuracy %
SGDM	10	770	70.45%
ADAM	10	770	62.48%

In deep networks, the choice of activation functions has a significant effect on training dynamics and task performance. The most successful and widely used activation function today is the linear rectified unit (ReLU). The table below shows the effect of the ReLU function compared to the tanh function.

Table 3: Classification accuracy for both ReLU and tanh

Function	Epochs	Max. iteration	Accuracy %
ReLU	10	770	70.45%
tanh	10	770	53%

After choosing the optimizer and the activation function and configuring our architecture, we run ConvNet for a number of different epochs. According to the table below, we notice that each time we increase the number of iterations the value of the accuracy becomes better [95].

Table 4: Evaluation parameters: ConvNet

Epoch	Max. iteration	Accuracy %
10	770	70.45%
30	2310	71.89%
40	3080	72.30%
50	3850	72.38%
100	7700	79.58%

5.2 Bayesian optimization

Bayesian optimization can have a great influence on model accuracy. To find optimal network hyper parameters and training options for CNN, we apply Bayesian optimization. This algorithm is well suited to optimize the hyper parameters of this classification model.
After preparing the data, we must choose the variables to optimize. These variables are the network section depth. The role of this parameter is to control the depth of the network. Initial learning rate and Stochastic gradient descent momentum. With initialization of the weights and the bias.

An important hyper parameter that influences the dynamics of the model is the number of examples of the training dataset called the batch size. It controls the accuracy of the estimate of the error gradient. Also, to evaluate the results when we run ConvNet, we define metric function to test the prediction performance of our model. This function selects 300 test images. It makes it possible to evaluate the network formed on these images and to calculate the proportion of images that the network has misclassified.

The figures below are an explanation of the convolutional operation with stride 2*2.

Figure 9: displays the training metrics at each iteration (10 epochs)

Figure 10: displays the training metrics at each iteration (40 epochs).

Figure 11: shows the frequency of dataset items (2 epochs).

Journal of Theoretical and Applied Information Technology
30th November 2021. Vol.99. No 22

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 12: a pie chart in which each slice of the pie chart represents a cell type (2 epochs).

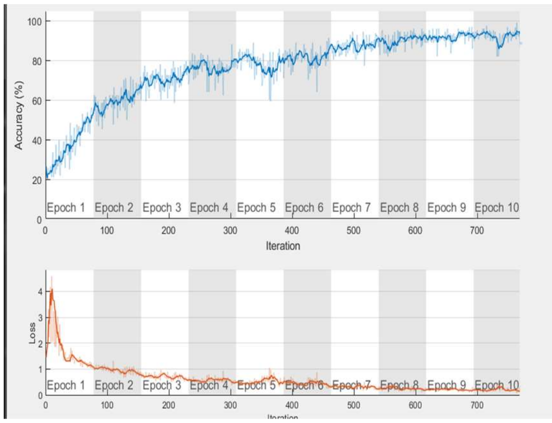

Figure 9. Training Progress specified as Plots value in training Options and start network training. This figure displays the training metrics at each iteration. Each iteration is an estimate of the gradient and an update of the network parameters (For 10epochs)

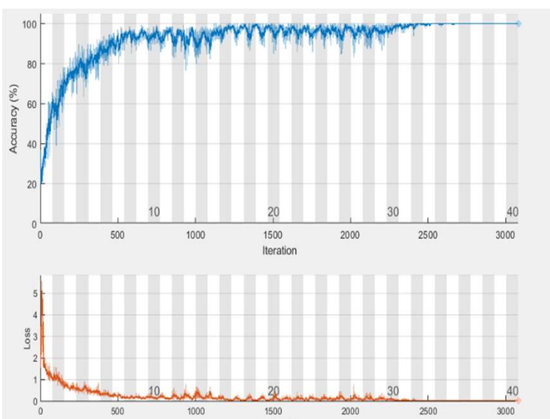

Figure 10. Training Progress specified as Plots value in training Options and start network training. This figure displays the training metrics at each iteration. Each iteration is an estimate of the gradient and an update of the network parameters (For 40 epochs)

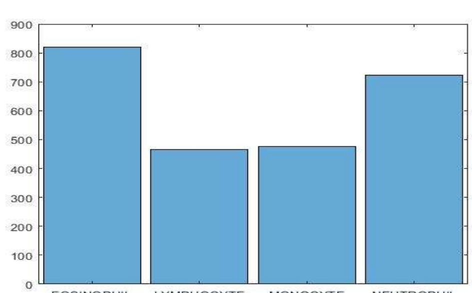

Figure 11. Histogram is a representation of statistical information showing the frequency of dataset items in successive intervals (4 cell types, for 2 epochs)

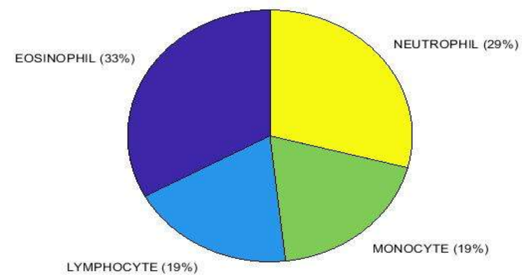

Figure 12: A pie chart using the dataset. Each slice of the pie chart represents a cell type (for 2 epochs)

Table 5: Evaluation Algorithm: Optimized-ConvNet before and after optimization

Epoch	Partitioning (train, test)	Accuracy (before optimization)	Accuracy (after optimization)
40	75%, 25%	72.30%	77.93%
100	75%, 25%	79.58%	93.37%

6. DISCUSSION

This study examined the feasibility of using deep learning to classify blood cell images into 4 classes. The results of our study show that ConvNet, the proposed approach, can be trained to extract the features and classify the input. In order to increase recognition rate, a Bayesian optimization of this architecture hyper parameters is also proposed.

Comparison with other CNN architectures and related works:

In this subsection, the performance of the proposed approach is compared to other convolutional neural networks mentioned in the table of related works in terms of classification accuracy. For example, G. Liang & all [78], using the same dataset with 12444 images.

Table 6: Comparison with other CNN architectures and related works

CNN models	Accuracy
CNN-RNN Xception LSTM	90.79%
CNN-RNN Inception V3-LSTM	87.45%
CNN-RNN ResNet50-LSTM	89.38%

CNN-RNN Xception-ResNet50-LSTM	88.58%
CNN Inception V3	84.08%
CNN-ResNet-50	87.62%
CNN Xception	88.70 %
Optimized-ConvNet	**93.37%**

Comparison with machine learning methods:

Previous works were based on Machine-learning techniques employed for blood cells images classification. For example, in [96], tested 5 different machine learning algorithms for the classification decision tree, Naive Bayes, Random Forest, KNN and SVM. They were used for 5 different datasets. All these datasets are trained and tested at different rates for each machine learning algorithm: worst, best, mean, standard deviation, median and mod values are calculated. In this section, we will present the most used techniques in each group K-Nearest Neighbors KNN and Support Vector Machine SVM [96].

The dataset is trained and tested at different rates for each machine learning algorithm. The values are calculated for 40 epochs and 100 epochs.

Table 7: Comparison of recognition performance for 40 epochs

Algorithm	Partitioning (train, test)	Accuracy
KNN	25%, 75%	74.90%
SVM	25%, 75%	77.19%
Optimized-ConvNet	**75%, 25%**	**77.93%**

Table 8: Comparison of recognition performance for 100 epochs

Algorithm	Partitioning (train, test)	Accuracy
KNN	75%, 25%	81.61%
SVM	75%, 25%	82.76%
SVM	67%, 33%	84.48%
Optimized-ConvNet	**75%, 25%**	**93.37%**

- **Advantages of the proposed method**

The suggested model can learn the features of blood cell images on its own, eliminating the necessity for manual feature extraction (no cytoplasmic/nuclear segmentation is required). Moreover, we use CNN performing by Bayesian optimization to improve the accuracy and the quality of the output. The experiment confirmed the validity of our proposal. Also, we applied the model on public databases to take the route of a comparison with other methods and to decide which is the most effective.

- **Limitations of the proposed method**

Despite its outstanding performance, our approach has a few flaws that could limit its practical use. In fact, the classification process is excessively slow. It can be resolved by increasing the performance of computer hardware or by optimizing the network's structure and reducing image size.

7. CONCLUSION

Due to its efficiency in diagnosing of blood-borne diseases including malaria and anemia at early stages, the convolutional neural network would be of great value to the medical diagnostic system used for detectors in blood cells.
In this work, we proposed an architecture based on Convolutional neural network for a multi-class task for detection of four classes of blood cells. It can automatically extract and classify the deep features embedded in cell image patches. We conducted several experiments to evaluate the performance of our model by performing Bayesian optimization, we are able to achieve 93,37% accuracy . In comparison to other current methods, our proposed strategy outperformed them in terms of classification on the blood cell dataset. We hope that this model can be used to develop medical-aided diagnostic systems for blood related diseases in the future.
As perspective, model improvement is still needed in term of accuracy improvements. We are considering to test the proposed methodology on other datasets and more complex datasets and for the treatment of different diseases.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 9: Summary of Deep Learning based Blood Cells Diagnosis

Year	Authors- Date	Number of images	Techniques	Performance (%)
2018	Shafique, S., & Tehsin, S. [77]	108 images + Data augmentation	AlexNet	Sensitivity 96.74% Specificity 99.03% Accuracy 96.06%
2019	G. Liang & all [78]	12444 images	CNN-RNN Xception LSTM CNN-RNN Inception V3-LSTM CNN-RNN ResNet50-LSTM CNN-RNN Xception-ResNet50-LSTM CNN Inception V3 CNN-ResNet-50 CNN Xception	Accuracy 90.79% Accuracy 87.45% Accuracy 89.38% Accuracy 88.58% Accuracy 84.08% Accuracy 87.62% Accuracy 88.70 %
2019	Yildirim, M., Çinar, A. [79]	9400 images	CNN-AlexNet CNN-ResNet50 CNNDensNet201 CNN-GoogleNet	Accuracy 82.16% Accuracy 80.02% Accuracy 83.44% Accuracy 75.21%
2019	Andrea Acevedo & all [80]	17092 images	CNN-Vgg16 CNN-InceptionV3	Accuracy 96% Accuracy 95%
2020	Sahlol & all [81]	10661 images	CNN -VGGNet+ SESSA	Accuracy 87.9%
2019	Loganathan, V [82]	434 images	Transfer learning +KNN classifier ANN classifier	Accuracy 88,04% Accuracy 54%
2020	Aliyu, Hajara Abdelkarim, & al. [83]	Over 9000 images	AlexNet	Accuracy 95.92% Sensitivity 77% Specificity 98.82% Precision 90%
2020	Abou El-Seoud, S., Siala, M., & McKee, G. [84]	10000 images	AlexNet	Accuracy 96.78%

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

REFERENCES

[1] X.-B. Jin, T.-L. Su, J.-L. Kong, Y.-T. Bai, B.-B. Miao and C. Dou, "State-of-the-Art Mobile Intelligence: Enabling Robots to Move Like Humans by Estimating Mobility with Artificial Intelligence," *Applied Sciences.,* pp. 8, 379, 2018.

[2] N. H. a. J. W. a. V. P. a. O. Gassmann, "Artificial intelligence and innovation management: A review, framework, and research agenda☆," *Technological Forecasting and Social Change,* pp. 162, 120392 , 2021.

[3] L. a. N. J. A. a. P. A. T. Drukker, "Introduction to artificial intelligence in ultrasound imaging in obstetrics and gynecology," *Ultrasound in Obstetrics \& Gynecology,* pp. 56, 498-505, 2020.

[4] U. a. S. A. S. a. D. Niyaz, "Advances in Deep Learning Techniques for Medical Image Analysis," in *Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC)*, 2018.

[5] F. W. S. W. X. J. J. T. D. Riccardo Miotto, "Deep learning for healthcare: review, opportunities and challenges," *Briefings in Bioinformatics,* vol. 19, p. 1236–1246, November 2018.

[6] A. L. G. D. H. Z. K. M. H. Holzinger, "Causability and explainability of artificial intelligence in medicine," *WIREs Data Mining Knowl Discov.,* vol. 9.4, p. e1312, 2019.

[7] T. S. I. U. a. P. C. S. i. a. C. N. N. I. J. o. C. I. S. 2. (. 2040007., "Traffic Sign Identification Using a Partially Cooperative Strategy in a Convolutional Neural Network," *International Journal of Cooperative Information Systems,* pp. 29, 2040007 , 2020.

[8] M. a. R. M. a. E. M. a. S. M. a. M. A. a. N. A. a. S. L. Wiering, "The Neural Support Vector Machine," 2013.

[9] R. N. M. D. R. e. a. Yamashita, "Convolutional neural networks: an overview and application in radiology," *Insights Imaging ,* vol. 9, p. 611–629, 2018.

[10] D. S. B. S. &. M. C. Waibel, "InstantDL: an easy-to-use deep learning pipeline for image segmentation and classification," *BMC Bioinformatics,* vol. 22, p. 103, 2021.

[11] S. a. S. S. a. R. A. a. S. P. K. Srivastava, "Deep learning for health informatics: Recent trends and future directions," 2017.

[12] F. M. a. S.-A. A. a. C. K. a. A. P. a. V. R. a. J. M. a. J. L. a. O. D. a. B. E.-W. a. K. B. a. N. Navab, "Hough-CNN: Deep learning for segmentation of deep brain regions in MRI and ultrasound," *Computer Vision and Image Understanding,* pp. 164, 92-102 , 2017.

[13] C. C. S. B. Jayita Saha, "Review of Machine Learning and Deep Learning Based Recommender Systems for Health Informatics," *Deep Learning Techniques for Biomedical and Health Informatics,* vol. 68, 2020.

[14] I. a. S. N. a. S. H. a. B. S. a. N. A. Singh, "Blood Cell Types Classification Using CNN," in *Bioinformatics and Biomedical Engineering, 8th International Work-Conference*, Granada, Spain, 2020.

[15] R. a. L. J. a. M. R. a. Z. W. N. W. Tomari, "Development of red blood cell analysis system using NI Vision Builder AI," *ARPN Journal of Engineering and Applied Sciences,* vol. 10, no. 19, pp. 8692--8698, 2015.

[16] M. K. C. L. E. F. A. M. L. S. Y. F. A. a. P. D. Lippeveld, "Classification of Human White Blood Cells Using Machine Learning for Stain-Free Imaging Flow Cytometry.," *Cytometry,* vol. 97, pp. 308-319, 2020.

[17] A. M. B. a. A. C. Mnassri, "Ga algorithm optimizing svm multi-class kernel parameters applied in Arabic speech recognition," *Indian Journal of Science and Technology ,* vol. 10.27, pp. 1-9, 2017.

[18] S. E. a. J. B. A. a. C. D. Jozdani, "Comparing Deep Neural Networks, Ensemble Classifiers, and Support Vector Machine Algorithms for Object-Based Urban Land Use/Land Cover Classification," *Remote Sensing,* vol. 11 , p. 1713, 2019.

[19] M. B. S. Chandra, "Survey on SVM and their application in image classification," *Int. j. inf. tecnol. ,* pp. 1-11, 2018.

[20] A. Tharwat, "Parameter investigation of support vector machine classifier with

kernel functions," *Knowl Inf Syst,* vol. 61, pp. 1269-1302, 2019.

[21] G. B. a. N. C. a. F. Amenta, "Machine learning in medicine: Performance calculation of dementia prediction by support vector machines (SVM)," *Informatics in Medicine Unlocked,* vol. 16, no. 100200 , 2019.

[22] Y. a. R. M. a. S. B. a. Y. A. M. a. W. W. Yu, "Multi-Image-Feature-Based Hierarchical Concrete Crack Identification Framework Using Optimized SVM Multi-Classifiers and D–S Fusion Algorithm for Bridge Structures," *Remote Sensing,* vol. 13, p. 240, 2021.

[23] H. e. a. Ohmaid, "Comparison between SVM and KNN classifiers for iris recognition using a new unsupervised neural approach in segmentation," *IAES International Journal of Artificial Intelligence ,* vol. 9.3 , p. 429, 2020.

[24] F. a. Y. Z. a. F. H. a. T. S. a. D. M. Emmert-Streib, "An Introductory Review of Deep Learning for Prediction Models With Big Data," *Frontiers in Artificial Intelligence,* vol. 3, p. 4, 2020.

[25] H. e. a. Ohmaid, "Comparison between SVM and KNN classifiers for iris recognition using a new unsupervised neural approach in segmentation.," *IAES International Journal of Artificial Intelligence,* vol. 9.3, p. 429, 2020.

[26] K. L. C. a. B. Y. Wang, "Spectral-similarity-based kernel of SVM for hyperspectral image classification," *Remote Sensing ,* vol. 12.13, p. 2154, 2020.

[27] M. a. B. S. K. a. N. D. a. M. A. Achirul Nanda, "A comparison study of kernel functions in the support vector machine and its application for termite detection," *Information,* vol. 9, no. 1, p. 5, 2018.

[28] D. T. a. X. T. a. X. L. a. X. Wu, "Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 26, no. 7, pp. 1088-1099, 2006.

[29] V. Jakkula, Tutorial on support vector machine (svm)., vol. 37, Washington State University, 2006.

[30] S. a. M. M.-I. Notley, "Examining the use of neural networks for feature extraction: A comparative analysis using deep learning, support vector machines, and k-nearest neighbor classifiers.," *arXiv preprint arXiv,* no. 1805.02294, 2018.

[31] M. e. a. He, "K Nearest Gaussian-A model fusion based framework for imbalanced classification with noisy dataset.," *Artif. Intell. Res. ,* vol. 4.2, pp. 126-135, 2015.

[32] A. Mohamed, "Comparative Study of Four Supervised Machine Learning Techniques for Classification," *International Journal of Applied Science and Technology ,* vol. 7, 06 2017.

[33] V. e. a. Hariraj, "Fuzzy multi-layer SVM classification of breast cancer mammogram images," *Int J Mech Engg Tech,* vol. 9.8 , pp. 1281-99, 2018.

[34] S. a. M. T. A. a. A.-Z. S. Albawi, "Understanding of a convolutional neural network," in *International Conference on Engineering and Technology (ICET)*, 2017.

[35] V. C. M. G. S. M. Suriya, "Design of Deep Convolutional Neural Network for Efficient Classification of Malaria Parasite," in *2nd EAI International Conference on Big Data Innovation for Sustainable Cognitive Computing*, 2021.

[36] S. A. Bini, "Artificial intelligence, machine learning, deep learning, and cognitive computing: what do these terms mean and how will they impact health care?," *The Journal of arthroplasty ,* vol. 33.8, pp. 2358-2361, 2018.

[37] W. Liu, "The Alexnet-ResNet-Inception Network for Classifying Fruit Images," *bioRxiv ,* 2020.

[38] H. a. V. A. a. F. D. Qassim, "Compressed residual-VGG16 CNN model for big data places image recognition," in *IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, 2018.

[39] F. e. a. Emmert-Streib, "An introductory review of deep learning for prediction models with big data.," *Frontiers in Artificial Intelligence,* vol. 3, p. 4, 2020.

[40] W. e. a. Liu, "A survey of deep neural network architectures and their applications," *Neurocomputing ,* vol. 234, pp. 11-26, 2017.

[41] T. L. a. S. F. a. Y. Z. a. P. W. a. J. Zhang, "Implementation of Training Convolutional Neural Networks," *CoRR,* vol. abs/1506.01195, 2015.

[42] P. e. a. Tiwari, "Detection of subtype blood cells using deep learning," *Cognitive Systems Research,* vol. 52, pp. 1036-1044, 2018.

[43] N. a. M. A. a. L. S. Prakash, "Mapping landslides on EO data: Performance of deep learning models vs. traditional machine learning models," *Remote Sensing,* vol. 12, no. 3, p. 346, 2020.

[44] M. a. F. S. a. O. A. Kraus, "Deep learning in business analytics and operations research: Models, applications and managerial implications," *European Journal of Operational Research,* vol. 281, no. 3, pp. 628--641, 2020.

[45] S. Mahapatra, Why Deep Learning over Traditional Machine Learning?, Towards Data Science, 2018.

[46] N. a. M. A. Akhtar, "Threat of Adversarial Attacks on Deep Learning in Computer Vision: A Survey," *IEEE Access,* vol. 6, pp. 14410-14430, 2018.

[47] L. a. O. W. a. W. X. a. F. P. a. C. J. a. L. X. a. P. M. Liu, "Deep learning for generic object detection: A survey," *International journal of computer vision,* vol. 128, no. 2, pp. 261--318, 2020.

[48] Y. a. B. Y. a. H. G. LeCun, "Deep learning," *nature,* vol. 521, no. 7553, pp. 436--444, 2015.

[49] T. a. F. S. a. Z. Y. a. W. P. a. Z. J. Liu, "Implementation of training convolutional neural networks," *arXiv preprint arXiv:1506.01195,* 2015.

[50] J. a. Z. D. a. C. G. a. L. N. a. X. D. Han, "Advanced Deep-Learning Techniques for Salient and Category-Specific Object Detection: A Survey," *IEEE Signal Processing Magazine,* vol. 35, no. 1, pp. 84-100, 2018.

[51] A. R. a. P. M. a. R. S. Pathak, "Application of deep learning for object detection," *Procedia computer science,* vol. 132, pp. 1706--1717, 2018.

[52] N. a. M. A. Akhtar, "Threat of Adversarial Attacks on Deep Learning in Computer Vision: A Survey," *IEEE Access,* vol. 6, pp. 14410-14430, 2018.

[53] P. P. a. S. S. Shinde, "A review of machine learning and deep learning applications," in *Fourth international conference on computing communication control and automation (ICCUBEA)*, 2018.

[54] Y. a. Y. J. Lu, "Automatic lip reading using convolution neural network and bidirectional long short-term memory," *International Journal of Pattern Recognition and Artificial Intelligence,* vol. 34, no. 01, p. 2054003, 2020.

[55] W. a. W. Z. a. L. X. a. Z. N. a. L. Y. a. A. F. E. Liu, "A survey of deep neural network architectures and their applications," *Neurocomputing,* vol. 234, pp. 11--26, 2017.

[56] W. a. K. K. a. Y. M. a. S. H. Yin, "Comparative study of CNN and RNN for natural language processing," *arXiv preprint arXiv:1702.01923,* 2017.

[57] M. S. a. R. S. David, "Comparison of word embeddings in text classification based on RNN and CNN," in *IOP Conference Series: Materials Science and Engineering*, vol. 1187, IOP Publishing, 2021, p. 012029.

[58] T. a. D. J. a. L. H. a. G. Y. Guo, "Simple convolutional neural network on image classification," in *IEEE 2nd International Conference on Big Data Analysis (ICBDA)*, 2017.

[59] D. a. E. R. a. F. E. a. M. D. a. T. A. a. W. C. Fourure, "Semantic segmentation via multi-task, multi-domain learning," in *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, 2016.

[60] S. a. M. T. A. a. A.-Z. S. Albawi, "Understanding of a convolutional neural network," in *International Conference on Engineering and Technology (ICET)*, 2017.

[61] A. a. C. S. a. N. N. Bakhshi, "Fast evolution of cnn architecture for image classification," Springer, 2020.

[62] M. a. B. J. J. a. F. D. R. Hussain, "A study on cnn transfer learning for image classification," in *UK Workshop on computational Intelligence*, 2018.

[63] L. a. M. O. Zaniolo, "On the use of variable stride in convolutional neural networks," *Multimedia Tools and Applications,* vol. 79, no. 19, pp. 13581--13598, 2020.

[64] A. a. S. A. a. Z. U. a. Q. A. S. Khan, "A survey of the recent architectures of deep convolutional neural networks," *Artificial Intelligence Review,* vol. 53, no. 8, pp. 5455--5516, 2020.

[65] Y. a. X. B. a. Z. M. a. Y. G. G. a. L. J. Sun, "Automatically designing CNN architectures using the genetic algorithm for image classification," *IEEE transactions on cybernetics,* vol. 50, no. 9, pp. 3840--3854, 2020.

[66] S. a. J. S. a. X. C. Yu, "Convolutional neural networks for hyperspectral image classification," *Neurocomputing,* vol. 219, pp. 88--98, 2017.

[67] X. a. W. L. a. Z. Y. a. S. Y. Zhang, "Graph-based place recognition in image sequences with CNN features," *Journal of Intelligent \& Robotic Systems,* vol. 95, no. 2, pp. 389--403, 2019.

[68] J. G. O. T. S. &. N. E. Guérin, "CNN features are also great at unsupervised classification," *arXiv preprint arXiv:1707.01700,* 2017.

[69] A. a. H. Y. a. C.-V. J. L. Craik, "Deep learning for electroencephalogram (EEG) classification tasks: a review," *Journal of neural engineering,* vol. 16, no. 3, p. 031001, 2019.

[70] G. a. L. G. a. F. L. a. T. B. a. G. P. Zhao, "Multiple convolutional layers fusion framework for hyperspectral image classification," *Neurocomputing,* vol. 339, pp. 149--160, 2019.

[71] G. a. P. S. a. P. I. a. K. Y. Kordopatis-Zilos, "Near-duplicate video retrieval by aggregating intermediate cnn layers," in *International conference on multimedia modeling*, 2017.

[72] J. a. W. Z. a. K. J. a. M. L. a. S. A. a. S. B. a. L. T. a. W. X. a. W. G. a. C. J. a. o. Gu, "Recent advances in convolutional neural networks," *Pattern Recognition,* vol. 77, pp. 354--377, 2018.

[73] I. a. C. M. Kandel, "A novel architecture to classify histopathology images using convolutional neural networks," *Applied Sciences,* vol. 10, no. 8, p. 2929, 2020.

[74] Z. a. R. D. Chen, "Research on Fast Recognition Method of Complex Sorting Images Based on Deep Learning," *International Journal of Pattern Recognition and Artificial Intelligence,* vol. 35, no. 06, 2021.

[75] V. H. a. R. E. J. a. o. Phung, "A high-accuracy model average ensemble of convolutional neural networks for classification of cloud image patches on small datasets," *Applied Sciences,* vol. 9, no. 21, p. 4500, 2019.

[76] P. a. G. R. a. P. P. a. T. A. a. R. A. Paygude, "IMAGE PROCESSING USING MACHINE LEARNING".

[77] S. a. T. S. Shafique, "Acute lymphoblastic leukemia detection and classification of its subtypes using pretrained deep convolutional neural networks," *Technology in cancer research \& treatment,* vol. 17, p. 1533033818802789, 2018.

[78] G. a. H. H. a. X. W. a. Z. L. Liang, "Combining convolutional neural network with recursive neural network for blood cell image classification," *IEEE Access,* vol. 6, pp. 36188--36197, 2018.

[79] M. a. C. A. C. Yildirim, "Classification of White Blood Cells by Deep Learning Methods for Diagnosing Disease," *Rev. d'Intelligence Artif.,* vol. 33, no. 5, pp. 335--340, 2019.

[80] A. A. S. M. A. P. L. &. R. J. Acevedo, "Recognition of peripheral blood cell images using convolutional neural networks," *Computer methods and programs in biomedicine,* vol. 180, p. 105020, 2019.

[81] A. T. a. K. P. a. E. A. A. Sahlol, "Efficient classification of white blood cell leukemia with improved swarm optimization of deep features," *Scientific reports,* vol. 10, no. 1, pp. 1--11, 2020.

[82] V. Loganathan, "Extraction of blood cell image classification using convolution neural network," *Int. J. Innov. Res. Adv. Eng,* vol. 6, pp. 2349--2163, 2019.

[83] H. A. a. R. M. A. A. a. S. R. a. R. N. Aliyu, "A deep learning AlexNet model for classification of red blood cells in sickle cell anemia," *Int J Artif Intell,* vol. 9, no. 2, pp. 221--228, 2020.

[84] M. H. S. G. M. Samir Abou El-Seoud, "Detection and Classification of White Blood Cells through Deep Learning Techniques," *International Journal of*

Online and Biomedical Engineering (iJOE), vol. 16, no. 15, 2020.

[85] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.

[86] S. e. a. Liu, "Classification of Ecological Data by Deep Learning," *International Journal of Pattern Recognition and Artificial Intelligence,* vol. 34, no. 13, 2020.

[87] M. a. E. Y. a. K. R. Akil Sr, "Computational aspects of deep learning models for detection of eye retina abnormalities," *Real-Time Image Processing and Deep Learning 2020},* vol. 11401, p. 1140109, 2020.

[88] M. Z. a. T. T. M. a. Y. C. a. W. S. a. S. P. a. N. M. S. a. H. M. a. V. E. B. C. a. A. A. A. a. A. V. K. Alom, "A state-of-the-art survey on deep learning theory and architectures," *Electronics,* vol. 8, no. 3, p. 292, 2019.

[89] Q. a. Y. L. T. a. C. Z. a. L. P. Zhang, "A survey on deep learning for big data," *Information Fusion,* vol. 42, pp. 146--157, 2018.

[90] J. R. a. C. P. M. England, "Artificial intelligence for medical image analysis: a guide for authors and reviewers," *American journal of roentgenology,* vol. 3, pp. 513--519, 2019.

[91] S. Postalcioglu, "Performance Analysis of Different Optimizers for Deep Learning-Based Image Recognition," *International Journal of Pattern Recognition and Artificial Intelligence,* vol. 34, 22 04 2019.

[92] J. a. Z. W. Gan, "Water Level Classification for Flood Monitoring System Using Convolutional Neural Network," *Proceedings of the 11th National Technical Seminar on Unmanned System Technology 2019,* pp. 299--318, 2021.

[93] S. a. R. M. a. A. F. a. M. R. a. S. A. Rahman, "Performance analysis of mAlexnet by training option and activation function tuning on parking images," in *IOP Conference Series: Materials Science and Engineering*, 2021.

[94] K. A. K. a. B. J. a. C. V. a. T.-R. I. a. N. T. K. Al-Dulaimi, "Classification of white blood cell types from microscope images: Techniques and challenges," *Microscopy science: Last approaches on educational programs and applied research (Microscopy Book Series, 8,* pp. 17--25, 2018.

[95] L. a. D. J. H. Muflikhah, "High performance of polynomial kernel at SVM Algorithm for sentiment analysis.," *JITeCS Journal of Information Technology and Computer Science,* vol. 3.2, pp. 194-201, 2018.

[96] A. a. T. M. Elen, "Classifying White Blood Cells Using Machine Learning Algorithms," *Uluslararası Muhendislik Arastirma ve Gelistirme Dergisi,* pp. 141-152, 31 01 2019.