Diagnosis of Malaria Infected Blood Cell Digital Images using Deep Convolutional Neural Networks

Mohammed AbdulAmeer Jabbar[1*], Abdulkareem Merhej Radhi[2]
Al-Nahrain University, Baghdad, Iraq

Abstract

Automated medical diagnosis is an important topic, especially in detection and classification of diseases. Malaria is one of the most widespread diseases, with more than 200 million cases, according to the 2016 WHO report. Malaria is usually diagnosed using thin and thick blood smears under a microscope. However, proper diagnosis is difficult, especially in poor countries where the disease is most widespread. Therefore, automatic diagnostics helps in identifying the disease through images of red blood cells, with the use of machine learning techniques and digital image processing. This paper presents an accurate model using a Deep Convolutional Neural Network build from scratch. The paper also proposed three CNN models each one trained on the Malaria RBC dataset with different architectures for handling the classification tasks. Furthermore, disadvantage of the traditional method of using transfer learning, and how to control model complexity to achieve better performance was discussed. The dropout regularization technique was used to avoid overfitting problems and minimize validation loss. Applying Data Augmentation technique to avoid the problem of small data in training of proposed models, which is a very common problem in medical dataset. Finally, removing noise in Malaria images using a Median blur filter, and studying how effects of that on training CNN models. According to the classification results, the proposed model achieved better classification results at accuracy 99.22 on the original Malaria RBCs dataset, and it has the best performance comparing with related work.

Keywords: Medical Image Classification, Convolutional Neural Network, Deep Learning, Malaria classification and detection, Automated diagnosis system

تشخيص الصور الرقمية لخلايا الدم المصابة بالملاريا باستخدام الشبكات العصبية الالتفافية العميقة

محمد عبد الأمير جبار عيسى[1*] ، عبد الكريم مرهج راضي[2]

جامعة النهرين, بغداد, العراق

الخلاصه

يعد التشخيص الطبي الآلي موضوعًا مهمًا خاصة في الكشف عن الأمراض وتصنيفها. الملاريا هي من أكثر الأمراض انتشارًا، حيث يوجد بها أكثر من 200 مليون حالة وفقًا لتقرير منظمة الصحة العالمية لعام 2016. عادةً ما يتم تشخيص مرض الملاريا باستخدام مسحات دم رقيقة وسميكة تحت المجهر. ومع ذلك، فإن التشخيص المناسب للمرض يكون صعباً للغاية خاصة في البلدان الفقيرة حين يكون انتشار المرض بشكل

واسع. لذلك تساعد التشخيصات التلقائية في التعرف على المرض من خلال صور الرقمية لخلايا الدم الحمراء
(RBC)، مع استخدام تقنيات التعلم الآلي ومعالجة الصور الرقمية. بهدف هذه البحث إلى تقديم نموذج دقيق
لتصنيف مرض الملاريا باستخدام الشبكة العصبية التلافيفية العميقة (DCNN) مبنية من الصفر. حيث تم
اقترح ثلاثة نماذج (CNN) تم تدريب كل منها على هيكلية مختلفة للتعامل مع مهمة تشخيص المرض. علاوة
على ذلك، تم التركيز على مساوئ الطريقة التقليدية لاستخدام نقل التعلم، وكيفية التحكم في تعقيد النموذج
لتحقيق أداء أفضل. ايضاً استخدمت تقنية (Dropout) لتجنب مشكلة (Overfitting) وتقليل فقدان معدل
التحقق من الصحة. أيضا استخدمت تقنية زيادة البيانات (Data Augmentation) للتعامل مع مشكلة
البيانات الصغيرة في عملية تدريب النماذج المقترحة ، وهي مشكلة شائعة جدًا في البيانات الطبية. أخيرًا، تم
إزالة الضوضاء في صور الملاريا باستخدام مرشح الوسيط Median blur ودراسة تأثير ذلك على عملية
تدريب نماذج الشبكة العميقة CNN. وفقًا لنتائج التصنيف، حقق النموذج المقترح أفضل النتائج بدقة 99.22
على مجموعة بيانات الملاريا الأصلية وكان لديه أفضل أداء مقارنة بالأعمال ذات الصلة.

1. Introduction

   Malaria is one of the common and dangerous diseases caused by a female Anopheles mosquito with parasites help. It is a very infectious disease for humans and other animals. It remains one of the most widespread infectious diseases of mankind with 216 million cases worldwide in 91 countries in 2016, according to the World Health Organization (WHO) [1]. Every year, hundreds of millions of blood films are examined for malaria, which involves a trained microscopy manually counting parasites and infected red blood cells (RBCs). Precise counts of parasites are not only essential for malaria diagnosis but it is also important for drug-resistant testing, measuring drug efficacy, and classifying the severity of the disease. However, Microscopic diagnostics are not standardized and strongly depend on microscopist's experience and skills. In low-resource settings, it is usual for microscopists to work in isolation, with no rigorous system in place that can ensure the upkeep of their skills and therefore the diagnostic quality. That leads to wrong diagnostic decisions in the field. A misdiagnosis involves unnecessary use of anti-malaria drugs for false-positive cases and going to suffer from their possible side effects, such as abdominal pain, nausea, diarrhea, and sometimes severe complications [1] [2]. Therefore, the automated system became an important method for assisted diagnosis. A new technique related to machine learning neural network called deep learning is developing recently to achieve better performance on classification tasks. One such form of deep-learning method is the Convolutional Neural Network (CNN). It is on top of image classification and recognition tasks, and it has the capability to simulate human vision [3].

1.1 Related Work

Many studies to detect malaria disease have been reported. For classification and detection problems, deep learning and other machine learning models have been used. Most of them are described below:

- Rosado, Da Costa et al. 2016 [4]. The authors of this paper introduced a technique for image processing and analysis, using supervised classification the main factor is the use of exclusively acquired microscopic images with low cost and accessible tools such as smartphones. This image used to detect Malaria parasite using an SVM classifier, the automatic detection achieved on white blood cells achieved sensitivity 98.2% and 72.1% of specificity.

- Dong, Jiang et al. 2017 [5]. In this paper the author evaluated Deep Convolution Neural Network (Transfer Learning TL) architecture, three models were trained including LeNet-5, AlexNet, and GoogLeNet, they were tested on the same dataset and compared with SVM. The dataset consists of 1032 infected RBC images and 1531 non-infected cells. All images were divided into two sets of approximately equal-sized. Also, the researchers used cross-

validation at 25% test. The result of all four methods reached classification accuracies above 90%, the SVM approach is less accurate than TL.

- Usha and Mallikarjunaswamy 2017 [6], Presented detection of Malaria using image processing techniques. Used images are acquired from thin blood smear slides and the captured color images are converted from (RGB) to gray-scale images, the noise reduction technique, contrast stretching for the image enhancement are employed. The infected red blood cell is segmented, classified using a support vector machine (SVM) with accuracy reached 90%.

- Devi, Roy et al. 2018 [7], the major issues presented in their work, are feature extraction, selection, and classification. The features such as prediction error, the co-occurrence of linear binary pattern, chrominance channel histogram, and R-G color are used. For (Support Vector Machine (SVM), k-Nearest Neighbors (KNN) and Naive Bayes) and hybrid classifier, obtained by combining the individual classifiers, is trained using the optimal feature set and the result is shown (sensitivity 95.86%, accuracy 98.5%, F-score 93.82%) achieved on the collected clinical database.

- Poostchi, Silamut et al. 2018 [8], this survey article has shown an update on the last development in automated malaria diagnosis with different methods for the feature extraction process and classification tasks using image analysis and machine learning. Deep learning is referred to in this article as the last tool was developed such as (DNN, CNN) to detect Malaria.

- Sadafi, Radolko et al. 2018 [9]. This study refers to the use of CNN in RBCs microscopic images on segmentation task; RBCs images are very useful for diagnosis of many diseases such as (Malaria, Anemia, etc.). For the learning process, the author used a different learning rate on different epochs the learning rate is then gradually reduced to allow the network to converge slowly. After training on 30 epochs the accuracy achieved at 93%.

- Sammy V. and et al. 2019 [10]. The study aimed to evaluate performance of CNN in several architecture using transfer leaning (TL) to detecting Malaria. They used TL models such as (Resnet, Googlenet and VGGne) and they achieved accuracy range from 90 to 96 % in classification of Malaria disease.

- Usha K. and et al. 2020 [11]. They used NIH Malaria dataset founded in website "Kaggle.com" to develop their work by utilize Discrete Wavelength Transformation (DWT). They presented image processing with Machine learning to optimize the classification of Malaria by using several methods for extracting features which are includes (Gray Level Co-occurrence Matrix (GLCM), Histogram of Oriented Gradients (HOG), Local Binary Pattern (LBP)) and classify the extracted features using SVM, the proposed model was achieved accuracy at 97.93%.

1.2 Deep learning

Deep learning (DL) is a new machine learning area that has recently become increasingly popular. DL algorithms can automatically learn features during the training process, and this is done in a much better way than using hand-coding to extract these features. Rather than crafting a set of rules and algorithms to extract features from the raw data [12]. Deep convolutional neural network (CNN), a class of artificial neural networks that have been a dominant method in computer vision tasks since the amazing results were shared in the object recognition competition known as the ImageNet Large Scale Visual Recognition Competition (ILSVRC) in 2012, is the most established algorithm among different deep learning models [13, 14]. CNN is a powerful learning algorithm for understanding image content that has shown exceptional performance in segmentation, classification, detection, and associated tasks [15]. The motivation of this method is to combine the feature extraction and classification processes to achieve a learning framework that overcomes the traditional method of features extraction (handcrafted). The use of CNN Architectures is a key

component of deep learning in the classification of images [16], it has demonstrated excellent efficiency in a lot of applications like image classification, speech recognition, object detection, and medical image analysis. The architecture of this method comes in several variants; however, they typically consist of convolutional and pooling layers, which are organized into modules. Either one or several fully connected layers, as in the standard neural feed-forward network. Modules are often stacked to form a deep model on top of each other. Figure 1, presents the typical CNN architecture for the classification of a toy image. The image is entered directly into the network, followed by a few stages of convolution and pooling. Finally, the last fully connected layer is the output of the class label [17].
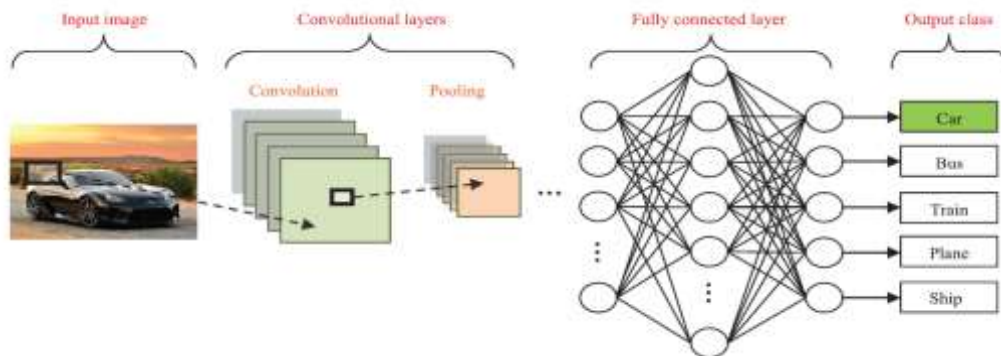
Figure 1-Typical CNN Image Classifications [17]

Overfitting is an unneglectable problem in deep CNNs, which can be effectively reduced by regularization. Overfitting refers to a situation where a model learns statistical regularities specific to the training set, i.e., ends up memorizing the irrelevant noise instead of learning the signal [18]. Dropout is a technique that helps to prevent overfitting and reduces validation error rates efficiently, the selection of which units to drop is random. In the actual system, each unit is acquired with a fixed probability p independent of other units, where p can be selected using a validation set or can simply be set to 0.5, which tends to be close to the optimum solution for a wide variety of tasks [19]. Many application domains do not have access to large data, such as medical image classification or medical diagnosis. Therefore, data augmentation is a solution to the problem of training a model with a small dataset. Data Augmentation represents a suite of techniques that improve the size and quality of training datasets, so which can build better deep learning models using them [20].

1.3 *Problem Statement*

The main problems in this research are identified as follows:

i. The Traditional methods of using transfer learning which is represented by using (Alexnet, Googlenet, Lenet...etc.) models, it's not necessarily lead to optimal results, especially in medical image classification tasks. Thus, computer-aided diagnosis is a system that simulates doctors or helps them to diagnose diseases. Those models were trained on (ImageNet, MNIST) datasets and it's completely different from the Malaria RBCs dataset.

ii. Classifying Malaria images according to their classes (non-infected cell, infected cell) with high sensitivity and more accurate results.

iii. Handle with small training data issue in Malaria dataset.

iv. Studying effects of removing noise with training deep models.

1.4 *Aim of this work*

The present research aims to build an accurate model using Deep Convolutional Neural Network from scratch by train several models with different architecture then select the best performance of classification for Malaria diagnosis using RBCs image dataset. Applying the Dropout regularization technique to avoid overfitting problem and minimize the testing error rate. Also, applying data augmentation technique to avoid the problem of small data in the

training of the proposed models, which is a very common problem in the medical dataset. Finally, removing noise in Malaria images using a median filter and studying how effects of that on deep models.

1.5 *Malaria images Dataset*

The malaria images dataset was combined from the group of researchers for the digital image of high magnification created by scanning an entire microscopic slide, which is available at the University of Alabama at Birmingham. This dataset was randomly selected from a large number of cell images and provided to pathologists at the University of Alabama. The dataset contains 2565 RBCs images that belong to two classes (the infected class with label 1, the non-infected class with label 0) [5].

2. Methodology

The main goal of this research is to apply an algorithm that has the ability to improve image classification performance based on deep convolutional neural network method to diagnose Malaria disease via an automated system. To design a CNN model from scratch, it is required to specify the best layers in the model architecture. Therefore, three models were proposed each one trained with different architecture. The best one will be select depending on validation accuracy. Figure 2, shows the proposed solution to design Malaria diagnosis models.
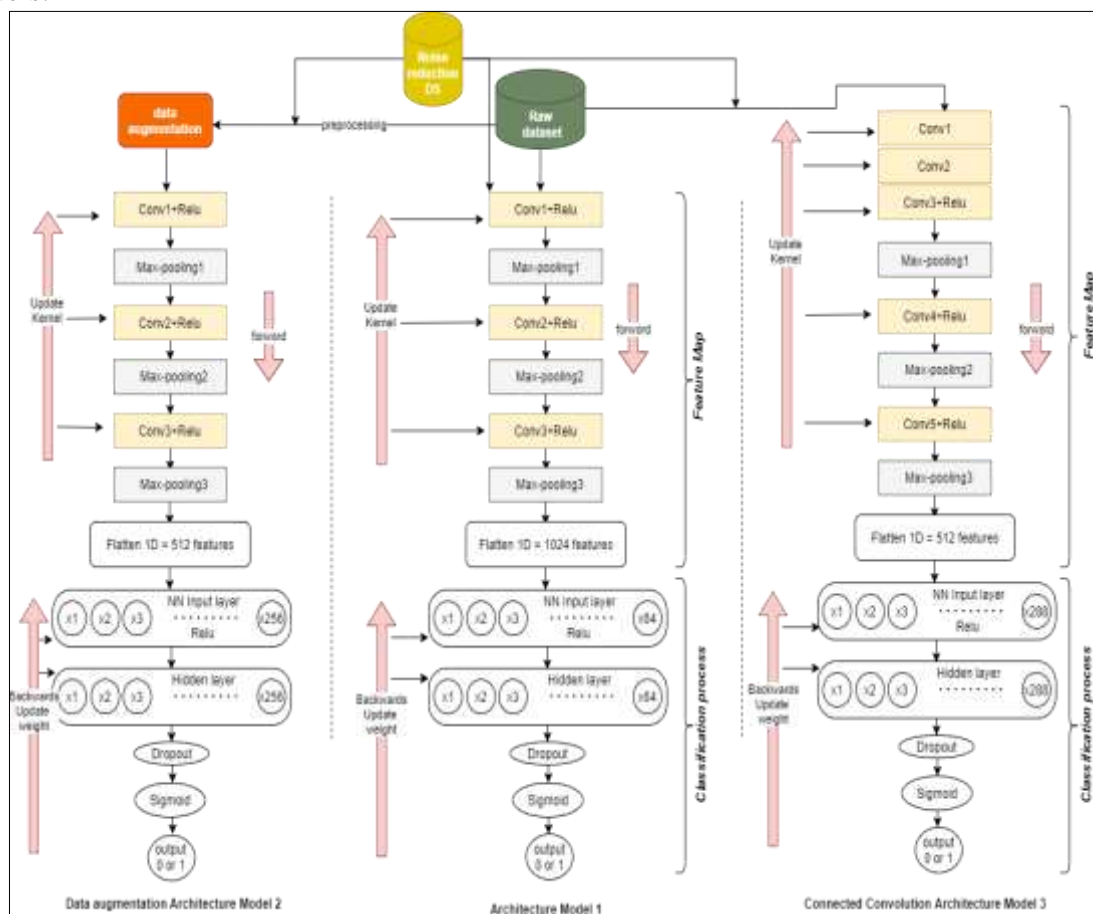

Figure 2-Block diagram of the proposed models.

2.1 Preprocessing

2.1.1 Removing noise from malaria RBC images:

The noise of Malaria RBC images was reduced using a Median blurred filter. According to this method, the noise was reduced at 3 * 3 which is the minimum value of noise removal in the image. The other sizes of the median filter such as 5*5, 7*7 are not applied because

images will be over smoothed and that effects quality of images. Figure 3 Smoothed Images (a) infected cell, (b) non-infected cell.

(a)   (b)
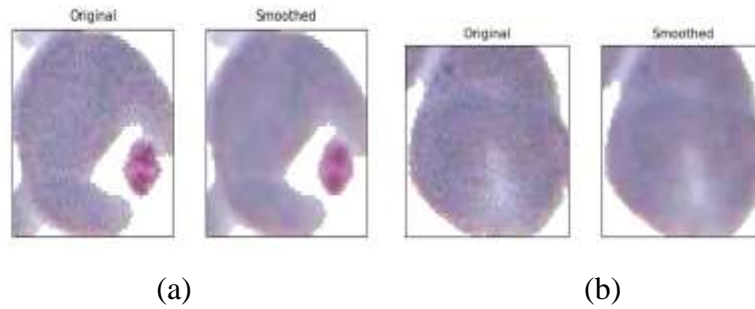
Figure 3- Smoothed Images (a) infected cell, (b) non-infected cell

2.1.2 Apply data augmentation technique:

Deep learning usually uses a large dataset for training; it is common to apply data augmentation to handle small dataset problem. Therefore, the data augmentation optimization technique aided to increase the Malaria RBCs dataset depending on the geometric transformation of images. The geometric transformation was proposed in Table 1 was applied to generate new augmented training set from the original one. Figure 4 shows the result of this technique on a single image.
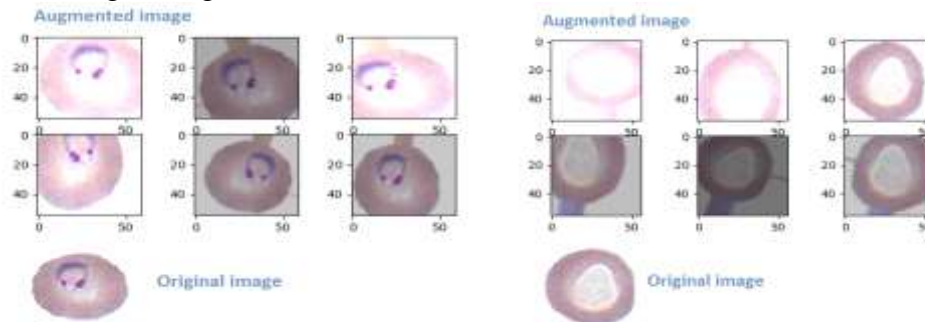
Figure 4-Augmented images for (left) infected class; (Right) non-infected class

2.1.3 Preparing dataset for training

Original RBCs Malaria images dataset contains 2565 samples belonged to two classes infected class with label 1, and non-infected class with label 0. The dataset was randomly splitting into two parts using the holdout method. The first part contains 80% of all images were used to training models, the second part contains a 20% validation set used for testing model performance.
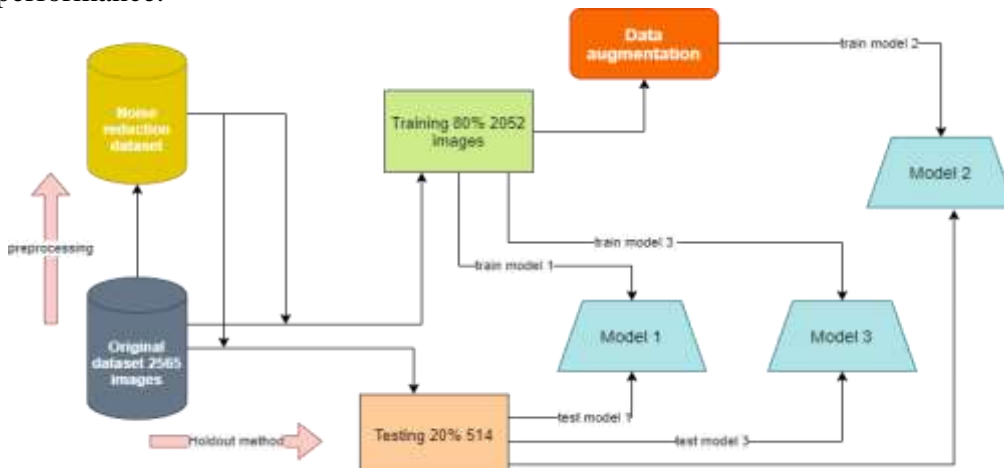
Figure 5-Datasets Splitting

2.2 Design Model 1 Architecture
This section involves building the basic layers of the proposed model 1. The first stage requires identifying how many convolution layers we need as a start, then the input image needs a fixed size.

Malaria images dataset is not of fixed sizes. If input images resized at 50 * 50 then, it required at least three convolution and pooling layers to extract he feature map. Let C be the convolution layer, Relu activation function, P Max-pooling layer with stride 2 that down sampling the feature map to half.

First (C1, Relu, P1) input 50*50 and the output 25 * 25, the second (C2, Relu, P2) input 24*24 and the output 12 * 12, the third (C3, Relu, P3) input 12*12 and the output 6 * 6.

This model was established with 14 layers. Starting with determining feature map for input images via the set of convolution and pooling, converted into one vector in flatten layer, and classified by fully connected layer. The next subsections contain details about these layers.

2.2.1 CNN Layers configuration and define parameters of model 1

I.   Feature map layers:

The feature map of input images has been determined utilizing three convolution layers with kernels and downsampling in the pooling layer. The first convolution layer contains 16 filters with size 3*3 with input image shape (height, width, and channel) as (50, 50, and 3); number 3 means used color images with RGB. The value of filters will be initialized randomly and updated according to the backpropagation rule. The activation function Relu, followed by a Max-pooling layer with 2*2 filter size and stride 2, which is downsampling the feature map by half. The second convolution layer consisted of 32 filters with size 5*5, the activation function Relu, then the Max-pooling layer with 2*2 filters size and stride 2. The third convolution layer has 64 filters with size 3*3, the activation function Relu, then the Max-pooling layer with 2*2 filter size and stride 2.

II. Flatten layer

The flatten layer involves taking all 2D output from the previous layers and convert it into one vector (1D); this vector is the input of the fully connected layer.

III.Fully connected layer

This layer is a traditional neural network with fully connected units which contains 1024 units ((note: that means takes all vectors in flatten layer you can pike any number and train model with it, we trained different vector from (1024, 512, 64) and discover if all 1024 have a robust feature). The output layer used sigmoid because the dataset has only two classes (infected (1), uninfected (0)).

2.2.2 Training and Testing of model 1

After reading the dataset, divide it into two subsets (train, test). The training set contains 2052 of Malaria RBC images was fitting to the proposed model. Binary cross-entropy has been used as a loss function; it gives a measure in the interval [0 - 1] of how separated two values. Binary cross-entropy can be defined as:

$$loss(\alpha, \acute{\alpha}) = -(a log(\acute{\alpha}) + (1-a)\log(1-\acute{\alpha})) \qquad \dots (1)$$

Where $\acute{\alpha}$ is the value returned by the model and α is the true label value. It's common to minimize $loss(\alpha, \acute{\alpha})$ for multiple images at the same time. The cost function is a function that we would like to minimize for a batch of images. Let $\acute{\alpha}i$ be the values returned by the model, and let αi be the true labels. S is the number of images in the batch. The mean of summation loss for a batch of images used as a Cost function and defined as:

$$Cost(\alpha, \acute{\alpha}) = \frac{1}{S}\sum_{i=1}^{S}[loss(\alpha i, \acute{\alpha}i)] \qquad \dots (2)$$

The goal of the optimizer is to minimize $Cost(\alpha, \acute{\alpha})$. Adam optimizer method has been used to minimize cost function. In practice, Adam was the best optimizer for the proposed model. The other methods such as (Stochastic gradient descent (SGD), root mean square prop (RMSprop)) were tried but, the convergence was slower compared with Adam and cannot

evaluate this model at 10 epochs. The hyperparameter of Adam optimizer initialized as: Learning rate = 0.001, B1=0.9 the value of first moment (mean), B2=0.999 the value of second moment (variance), and epochs = 10, batch size = 32.

2.2.3 Model 1 Regularization Using Dropout Technique

The dropout regularization technique was used to prevent overfitting of the proposed model. This technique works by a set of random units to zero in a fully connected hidden layer. The dropout rate is a value that refers to how many units are dropped temporarily. For example, 0.5 dropout rate refers to half of all units in the FC hidden layer will be dropped. In the proposed model, the dropping rate = 0.5 has been utilized to prevent overfitting and minimize validation loss, after adding the dropout layer, the total layers of model 1 became 15 layers.

2.3 Experiment Training with data augmentation (Model 2)

This experiment involved trials to limit the issue of small training dataset in Malaria RBCs images; a Data augmentation technique was proposed to handle this issue. This technique is effective in creating more training data samples by applying geometric transformation of each image. There are two ways to apply data augmentation on the input dataset. The first, use all images in available data and generate more data samples, then re-split the new dataset into a subset using cross-validation methods. The second, apply this technique only with the training set, and before initialize the training process. The second way has been selected to generate more data samples for training sets. This experiment was trained on the same architecture of proposed model 1. The number of trainable parameters was increased to 154113 sample after applying data augmentation, which needs to add a few modifications on proposed model 1 architecture which includes:

1- Instead of using 64 features in the first input layer of classification stage (layer 12), tried to increase the input unit to (256, 512) of the network depending on the previous layer (flatten).

2- Minimize the number of filters at convolution layer 3 to32.

2.4 Design Connected Convolution Architecture (Model 3)

This experiment involves applying connected convolution layers. These layers can extract more robust features from the first operation of convolving filters on input images. According to model 2 summary, first convolution has input image size (48, 48, 16) extracted features by convolving 16 filters for each color (R, G, B), followed by activation (Relu) to neglects negative values of convolving 16 filters on the input image. After this operation, the Max-pooling layer extracted maximum values with another filter at size (2, 2), stride 2 which is down sampled the convolution outputs to half. The input RBCs images have a proposed shape at (50,50,3) so tried to add connected convolution layers at this stage were further exploit the original size of an image, and before the convolved image downsampling by max-pooling layer. The proposed model 3 was trained and tested on the original dataset without use of data augmentation. All three model's architecture was proposed in previous experiments was trained and evaluate at 10 epochs; so, tried to increase the number of iterations with saving the best weight at 50 epochs.

3. Result and Analysis

3.1 Implementation issues

In the present work, the implementation issues were used for hardware workstation laptop with Core i7 CPU (4 core, 8 CPU, 1600 thread), the GPU Nvidia Quadro 2 G dedicated, 16 RAM. The software issues Python 3.7 version, Keras on TensorFlow backend, anaconda 4 environment, open cv library.

3.2 Experimental results of Architecture model 1

Model 1 was established with 14 layers, the first 9 layers involved with extracting feature map, then converting it to one vector at (1024) in layer 10, and the last 4 layers related with classification stage. The classification stage (Fully Connected) consisting of input, hidden

layer, and Sigmoid output layer. The input image size was determined at 50 * 50 with three components R, G, B. Table 1 shows model 1 architecture.

Table 1-Model 1 Architecture

Layer name	Number of kernels, stride	Output image shape
Input image	-	(50,50,3)
Conv 1	16	(48,48)
Relu	-	(48,48)
Max- pooling1	2,2	(24,24)
Conv 2	32	(22,22)
Relu	-	(22,22)
Max- pooling 2	2,2	(11,11)
Conv 3	64	(9,9)
Relu	-	(9,9)
Max- pooling3	2,2	(4,4)
Flatten	-	1024 vector
Input NN	-	64 unit
Relu	-	64 unit
Hidden	-	64 unit
Sigmoid	-	1 unit

From Table 1 the number of kernels increased at 16, 32, 64 which are suitable for the first design but, the output Flatten layer 1024 vector. Therefore, tried to choose at random from (1024, 512, 64) to determine how many robust features in 1024 vector that able to distinguish two classes and reduce the model complexity. The confusion matrix (CM) was computed on the test set which contains 514 samples belonging to two classes (infected with the class label "1", non-infected "0"). Table 2 shows CM result of model 1 at each vector, while Table 3 depicts the results of selecting the best vectors.

Table 2-Model 1 CM with three different vectors

Model 1 Flatten (V)	True class	Predicted class	
		Positive (1)	Negative (0)
1024 V	Positive (1)	224	22
	Negative (0)	4	264
512 V	Positive (1)	229	17
	Negative (0)	3	265
64 V	Positive (1)	231	15
	Negative (0)	0	268

Table 3- result of selecting best vectors

# Input FC Vector	# of Trainable Parameter	Test loss	Test accuracy %
1024	1,074,209	0.1513	94.94
512	548,897	0.1414	96.11
64	89,249	0.1153	97.08

Depending on the result shown in Table 3 1024, 500 vectors had matched features map and a large number of trainable parameters. The 64 vectors are better testing accuracy and little complexity than model 1. The next step tried to identify the best learning rate of Adam optimization method at vector 64 and 10 epochs, Table 4 showing the result of the learning rate.

Table 4- result of the different learning rate

| Learning rate of Adam	Testing loss	Testing accuracy %
0.01	0.6886	52.33
0.001	0.1153	97.08
0.0001	0.1948	92.22

According to the result shown in Table 4, the best learning rate is 0.001 which is a default rate of Adam and was suitable to update kernel and weight at 10 epochs.

After selecting the best vectors and learning rate of model 1. The dropout regularization technique was applied to avoid overfitting and minimize validation loss. Model 1 was retrained with the saved best result at 10 epochs Table 5 shows the result of each 10 epochs with dropout at rate 0.5.

Table 5-Results of with/without Dropout at 10 epochs

Apply dropout	Test loss	Test accuracy
No	0.1153	97.08
Dropout (0.5)	0.0918	97.64

The graphs of accuracy and loss between training and testing of model 1 after applying dropout layer shown in Figure 6.
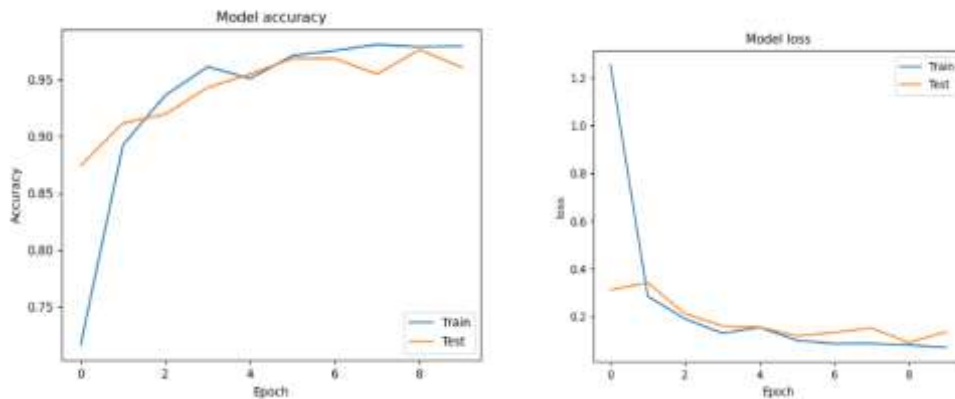
Figure 6-Graph of loss and accuracy of model 1 at 10 epochs

3.3 Experimental result of Architecture model 2
Model 2 was trained with data augmentation. The architecture of model 2 is similar to model 1 with few modifications to minimize the total vector in the flatten layer. Table.6 shows model 2 architecture.

Table 6-Model 2 Architecture

Layer name	Number of kernels, stride	output image shape
Input image	-	(50,50,3)
Conv 1	16	(48,48)
Relu	-	(48,48)
Max- pooling1	2,2	(24,24)
Conv 2	32	(22,22)
Relu	-	(22,22)
Max- pooling 2	2,2	(11,11)
Conv 3	32	(9,9)
Relu	-	(9,9)
Max- pooling3	2,2	(4,4)
Flatten	-	512 vectors
Input NN	-	256 unit
Relu	-	256 unit
Hidden	-	256 unit
Dropout	0.5	128 unit
Sigmoid	-	1 unit

From Table 6, this model is established with 15 layers. The First 9 layers related to extracting the feature map, then converted the output images into one vector (512 vectors) at flatten layer 10, and the last 5 layers involved with the classification process. Also, tried to select between two Flatten vectors (512, 256) to identify the best feature map and balancing the complexity of model 2, the dropout regularization method was implemented at rate 0.5. The

scope of this experiment is to train model 2 with data augmentation to obtain better classification accuracy of Malaria RBCs images.

The confusion matrix of this experiment was also computed on the test set with 514 samples. Table 7 shows CM of model 2 with each vector at epochs=10, Table 8 shows the classification result of model 2 including (recall, precision).

Table 7-Model 2 CM at 10 epochs

Model 2 Flatten (V)	True class	Predicted class	
		Positive (1)	Negative (0)
256 V	Positive (1)	231	15
	Negative (0)	2	266
512 V	Positive (1)	234	12
	Negative (0)	6	262

Table 8- model 2 classification result

# of Vectors	Test loss	Test accuracy %	Recall	Precision
256	0.1187	96.69	0.9390	0.9914
512	0.1145	96.50	0.9512	0.9750

Through the results shown in Table 8, both vectors have similar results, but vector 256 achieves the least complexity of Model 2, and thus the accuracy may improve after increasing the number of iterations. Compared with the first model, results did not improve after increasing the training data using data augmentation at 10 epochs.

3.4 Experimental results of connected convolution architecture model 3

The aim of this experiment is further exploited for the original size of input images. The connected convolution was applied in the first input image size at (48, 48). The proposed model was established with 17 layers. The First 12 layers are related to the extract feature map, and the last 5 layers involve in the classification process. Table 9 shows the architecture of model 3.

Table 9-Model 3 Architecture

Layer name	Number of kernels, stride	output image shape
Input image	-	(50,50,3)
Conv 1	16	(48,48)
Conv 2	32	(46,46)
Conv 3	32	(44,44)
Relu	-	(44,44)
Max- pooling1	2,2	(22,22)
Conv 4	32	(18,18)
Relu	-	(18,18)
Max- pooling 2	2,2	(9,9)
Conv 3	32	(7,7)
Relu	-	(7,7)
Max- pooling3	2,2	(3,3)
Flatten	-	228 vectors
Input NN	-	228 unit
Relu	-	228 unit
Hidden	-	228 unit
Dropout	0.5	114 unit
Sigmoid	-	1 unit

Model 3 was trained on the original Malaria RBCs dataset at 10 epochs, Table 10 shows the confusion matrix and the classification result.

Table 10-Model 3 CM at 10 epochs

True class	Predicted class		Test loss	Test accuracy %
	Positive (1)	Negative (0)		
Positive (1)	232	14	0.08 27	97.28
Negative (0)	0	268		

According to the classification result shown in Table 10, model 3 achieved a better classification result on the original Malaria RBCs dataset. The next stage is to increase number of epochs at 50 and evaluate model's performance.

3.5 Evaluate models' performance at 50 epochs

This stage aimed to increase training iteration at (50 epochs). In the province experiments, all model architectures were trained at 10 epochs which are considering small number of training iteration. Adam optimizer cannot obtain the best weights and kernel update at epochs =10. But, small training iteration for all models was good for identifying errors in designing architectures of proposed models. The confusion matrix of the proposed models was computed on the testing set which contains (514 RBCs images) belong to two classes (infected 1, non-infected 0). Table 11 shows the confusing matrix of each model at 50 epochs.

Table 11-Confusion matrix at 50 epochs

Model	True class	Predicted class	
		Positive (1)	Negative (0)
Model 1	Positive (1)	232	14
	Negative (0)	0	268
Model 2	Positive (1)	238	8
	Negative (0)	5	263
Model 3	Positive (1)	242	4
	Negative (0)	0	268

Depending on confusion matrix results that present in Table 11, the evaluation metrics (loss, accuracy) shown in Table-12

Table 12-Classification result at 50 epochs

Model name	Testing loss	Accuracy %
Model 1	0.0931	97.28
Model 2	0. 0894	97.47
Model 3	0. 0487	99.22

From the result shown in Table 12; the architecture of model 3 achieved the best test accuracy and test loss compared with others. This model can classify Malaria images depending on the feature map extracted by connected convolution and without using the data augmentation technique. Table 13, shows other metrics for further evaluation of model performance. These metrics include sensitivity (recall), specificity (precision), F1-score.

Table 13-Other classification metrics result

Model name	Recall	Precision	F1-score
Model 1	0.9431	1	0.9707
Model 2	0. 9675	0. 9794	0.9734
Model 3	0.9837	1	0.9917

According to the result shows in Table 13, model 3 recorded a higher score than others. Figure 7 shows the graphs of accuracy and loss between training and testing of model 3.
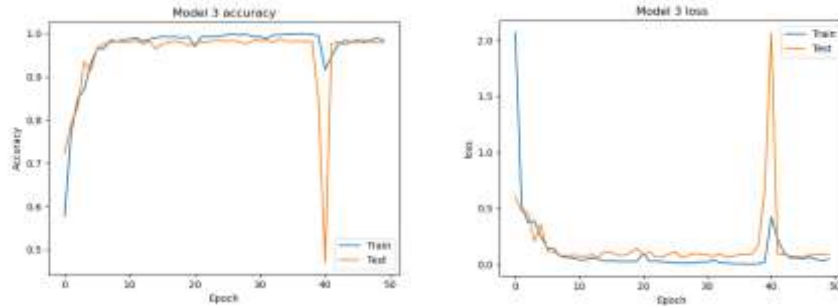
Figure 7-graph of loss and accuracy of model 3 at 50 epochs

From figure 7, note the pointer in epochs 40 are not stable and the gap between training and testing is large, which indicates the model is going through overfitting problem. Also, it can be seen the next epochs after 40 will be stable and the gap is very small. That is because the effectiveness of the dropout regularization technique to prevent overfitting. While model 2 was trained using data augmentation as shown in figure 8 the gap is small and the pointers are stable which indicates model 2 was avoiding overfitting because of increasing data samples by using data augmentation.
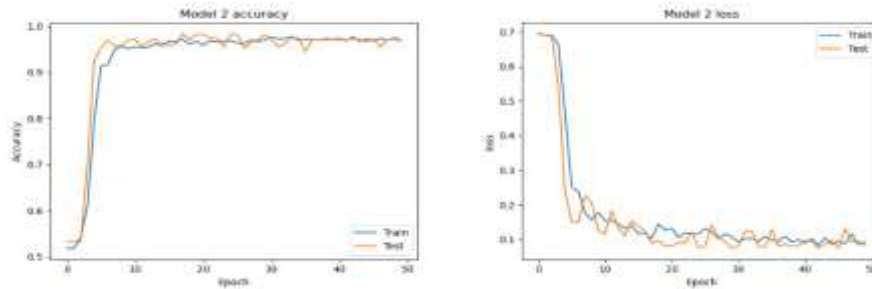
Figure 8- Graph of loss and accuracy of model 2 at 50 epochs

3.6 Training with noise reduction result
The purpose of this experiment is to attempt to reduce noise in the original dataset of malaria and study how effects of that on training the proposed models. Therefore, the method of preprocessing which is related to removing noise by applying median blur filter was used to create a new dataset. The noise reduction dataset was trained on three proposed models at 50 epochs and its results are shown in Table 14.

Table 14-Confusion matrix with noise reduction DS

Model name	True class	Predicted class	
		Positive (1)	Negative (0)
Model 1	Positive (1)	231	15
	Negative (0)	0	268
Model 2	Positive (1)	233	13
	Negative (0)	1	267
Model 3	Positive (1)	240	6
	Negative (0)	1	267

Table 15-classification result with noise redaction at 50 epochs

Model name	Testing loss	Accuracy %
Model 1	0.1033	97.08
Model 2	0. 0953	97.28
Model 3	0.0688	98.64

As the results are shown in Table 15 all models were trained and tested on a noise reduction dataset. The classification accuracy did not improve compared with results shown in Table 12 at the same epochs when models were trained without removing noise.

3.7   Comparing with related work on the same dataset

This section involves with comparing the results presented in Table 4.12 with other related work on the same dataset. Table 4.16 shows the proposed models compare with other models.

Table 16-comparing with related work on the same dataset

Model name	Used data augmentation	Accuracy
Proposed model 1	No	97.28
Proposed model 2	Yes	97.47
Proposed model 3	No	99.22
TL with LeNet-5 [5]	No	96.18
TL with AlexNet [5]	Yes	95.79
TL with GoogLeNet [5]	yes	98.13
SVM with hu- moment features [5]	-	91.66

According to the comparison results shown in Table 16, the proposed model 3 achieved the best classification accuracy on the original Malaria dataset and without using data augmentation.

3.8   Analysis

From the three models designed, the first model extracts a feature map at 1024 vectors. Through the classification results reported in Table 3, vector 64 obtained the best accuracy result of classification for Malaria images and also achieved a balance between model complexity and accurate results.

When determining the best learning rate mentioned in Table 4 for Adam's algorithm, learning rate 0.001 was the best choice for updating weights and filters as well. Thus, this rate will be used for all subsequent experiments.

After selecting best vector from flatten layer and best learning rate and applied the dropout technique, the proposed model 1 achieved 97.6 accuracy for classification Malaria RBCs dataset. But this model misclassified with 14 samples at FN rate as shown in Table 2; the reason could be explained by the fact that the training data set was small, or the first model needed to be modified in the architecture. Consequently, the second model proposed with increasing the training data set using the data augmentation technique with a slight modification of the model's architecture.

The second model (Model 2) was trained using the data augmentation technique on the training set. This proposed technique aims to increase the input images using geometric transformations as shown in Figure 2. Model 2 extract feature map at 512 vectors, according to result mentioned in Table 8; the 256 vectors of flatten layer was achieved classification accuracy 96.69 with less model complexity for trainable parameter and misclassified at FN = 15, FP = 2 samples. While 512 vectors are achieved classification accuracy at 96.5 with more complexity and misclassified at FN = 12, FP = 6 samples. Comparing 512, and 256, 256 was higher accuracy and may be the result of misclassification will be improved when increasing number of epochs. Overall, the data augmentation technique did not improve the performance

of the second model compared to the first model in 10 epochs. Through this experience, it can be diagnosed that the training data is not little, but the second model needs to be modified in the proposed architecture.

The third model modified used connected convolution architecture. These layers can extract more features in the original size of the input image and before Max-pooling 1 downsampling to half. The proposed model 3 extract feature map at 512 vectors and trained on the original Malaria RBCs dataset at 10 epochs. According to the result mentioned in Table 10, this model achieved the best classification accuracy 97.28 and misclassified with FN = 14, FP = 0 of samples.

In the next stage, all proposed model was retrained with increase number of epochs at 50. As a comparison of evaluation results was mentioned in Table 12 the architecture of model 3 was achieved the best classification accuracy 99.22 with minimum loss at 0. 0487, and misclassified with only FN = 4 and FP = 0 of test samples according to confusion matrix results mentioned in Table 11. The following experiment related to studying de-noise Malaria input images and the effect of training the proposed models.

According to removed noise results (shown in Figure 1), the de-noise images were used to retraining the proposed models at 50 epochs. As the results were mentioned in Table 14. The results indicate that removing noise from the input images did not improve the classification performance. This may be due to the difficulty of extracting features in convolution layers when the images are smoothed.

The last section is related to compare our proposed models with related work on the same dataset. According to the results mentioned in Table 16, model 3 achieved better classification accuracy in the Malaria RBCs dataset.

The results also showed that the use of transfer learning does not necessarily have the optimal classification accuracy, especially in diagnosing medical images. The GoogleNet model was trained on the ImageNet dataset and used 256 *256 image sizes, the number of trainable parameters 5,975,602 obtained with 22 convolution layers. While proposed model 3 has 50 * 50 image size and 150,025, trainable parameter, 6 convolution layers. It is thus less complex than a GoogleNet and its design was appropriate and further accurate for diagnosis Malaria images. So, we were focusing on the start of the balance between model complexity and good results.

4.   Conclusion

The convolutional neural network is a very efficient method that can be used to accurately diagnose the classification and detection of malaria. To achieve accurate results in CNN models, it is necessary to select the best layers in the model architecture. The selection of the kernel size and number in the convolution layer depends on the classification task. Adam's optimization method was faster in update weight and kernel with a learning rate of 0.001. The dropout regularization technique was an efficient method to avoid the overfitting problem and minimize testing loss. The data augmentation technique can be a help to solve the problem of a small training dataset. Building several models with different architectures was good for selecting the best one and specifying errors in the design CNN model. Applying connected convolution in the first layers was an effective method to extract more robust feature map. It can be more accurate than using data augmentation. The use of transferred learning conveyed by (Googlenet, Alexnet, etc.) did not lead to ideal classification accuracy because these models contain many trainable parameters, more complex, and not suitable for the classification problem. Which affects the extraction of a less accurate feature map and thus affects the classification results. Removing noise in Malaria RBCs images decreased classification accuracy. This indicates that it is difficult to extract the features map when the image is smooth and because most convolution features are edge detection.

References
[1] "World Malaria Report, " World Health Organization, Geneva, 2017.
[2] W. H. Organization, "*Malaria microscopy quality assurance manual-version 2*," World Health Organization, 2016.
[3] D. S. Kermany *et al.*, "Identifying medical diagnoses and treatable diseases by image-based deep learning," vol. 172, no. 5, pp. 1122-1131. e9, 2018.
[4] L. Rosado, J. M. C. Da Costa, D. Elias, and J. S. J. P. C. S. Cardoso, "Automated detection of malaria parasites on thick blood smears via mobile devices," vol. 90, pp. 138-144, 2016.
[5] Y. Dong *et al.*, "Evaluations of deep convolutional neural networks for automatic identification of malaria infected cells," in *2017 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*, 2017, pp. 101-104: IEEE.
[6] D. Usha and M. Mallikarjunaswamy, "Detection of Malaria Based on the Blood Smear Images Using Image Processing Techniques,"
[7] S. S. Devi, A. Roy, J. Singha, S. A. Sheikh, and R. H. Laskar, "Malaria infected erythrocyte classification based on a hybrid classifier using microscopic images of thin blood smear," *Multimedia Tools and Applications,* vol. 77, no. 1, pp. 631-660, 2018.
[8] M. Poostchi, K. Silamut, R. J. Maude, S. Jaeger, and G. Thoma, "Image analysis and machine learning for detecting malaria," *Translational Research,* vol. 194, pp. 36-55, 2018.
[9] A. Sadafi, M. Radolko, I. Serafeimidis, and S. Hadlak, "Red Blood Cells Segmentation: A Fully Convolutional Network Approach," in *2018 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Ubiquitous Computing & Communications, Big Data & Cloud Computing, Social Computing & Networking, Sustainable Computing & Communications (ISPA/IUCC/BDCloud/SocialCom/SustainCom)*, 2018, pp. 911-914: IEEE.
[10] S. V. Militante, "Malaria Disease Recognition through Adaptive Deep Learning Models of Convolutional Neural Network," in *2019 IEEE 6th International Conference on Engineering Technologies and Applied Sciences (ICETAS)*, 2019, pp. 1-6: IEEE.
[11] U. Kumari, M. M. Memon, S. Narejo, and M. Afzal, "Malaria Disease Detection Using Machine Learning."
[12] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*, MIT press Cambridge, 2016.
[13] O. Russakovsky *et al.*, "Imagenet large scale visual recognition challenge," vol. 115, no. 3, pp. 211-252, 2015.
[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097-1105.
[15] X. Liu, Z. Deng, and Y. J. A. I. R. Yang, "Recent progress in semantic image segmentation," vol. 52, no. 2, pp. 1089-1106, 2019.
[16] T. Guo, J. Dong, H. Li, and Y. Gao, "Simple convolutional neural network on image classification," in *2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)(*2017, pp. 721-724: IEEE.
[17] W. Rawat and Z. J. N. c. Wang, "Deep convolutional neural networks for image classification: A comprehensive review," vol. 29, no. 9, pp. 2352-2449, 2017.
[18] R. Yamashita, M. Nishio, R. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," Insights Imaging 9 (4), 611–629 (2018)," ed.
[19] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. J. T. j. o. m. l. r. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," vol. 15, no. 1, pp. 1929-1958, 2014.
[20] E. J. J. a. p. a. Bjerrum, "SMILES enumeration as data augmentation for neural network modeling of molecules," 2017.