WEIGHT OPTIMIZATION ALGORITHMS FOR DENOISING MEDICAL IMAGES USING CONVOLUTIONAL NEURAL NETWORK

FEMI[a*], SHENBAGALAKSHI[b]

[a]*University College of Engineering Nagercoil, TamilNadu.*

[b]*SKN Sinhgad Institute of Technology & Science, STES Campus, Lonavala, Maharastra.*

ABSTRACT

A novel combination of algorithms along with convolutional neural networks (CNNs) to denoise medical images is proposed in this paper. CNN is a type of deep learning model that specializes in retrieving information from input images instantly and capability to reduce the need for expert knowledge when extracting and selecting features. Hyper parameters like activation functions can have a direct impact on the model's performance in CNN and hence the paper, Improved Rectified Linear Units (I-ReLU)-CNNs are proposed for denoising the Medical Images(MI). In addition, the system also uses an encoder-decoder structure to save the most significant features of medical images while discarding the ones that are not required and by using a residual learning technique, the network is trained from start to end. Existing denoising techniques extract the clean image directly rather than learning the noise from the noisy image, where the denoised images are acquired through a proper learning method and the improved CNN layer. Furthermore, to optimize the weight ratio of CNN, different combinations of optimization algorithms are analysed and the network layer weight is optimized to get a denoised and clear image of MI. The proposed method not only denoises the MI but also completely removes the unwanted artifacts that would interfere with the blood cell counting stage. Extensive simulation is carried out and the noise is effectively removed and the mean square error (MSE) is reduced to zero. Thus the proposed optimized Denoising Convolutional Neural network (DnCNN) method for MI processing achieves excellent performance with reduced MSE and the desirable performance parameters.

Keywords: *Blood Cell Image (BCI), Mean Square Error, DnCNN, Residual learning, Improved ReLU.*

* For Correspondence *femiraj.eee@gmail.com*

1. INTRODUCTION

In the realm of biomedicine, blood cell analysis is the initial step in identifying many illnesses. A physician initial suggestion for detecting diseases is a Complete Blood Cell Count (CBCC). The images used to analyse blood cells can help evaluate and diagnose a patient's multiple health issues. Unusual high or low blood cell counts may indicate the presence of a number of illnesses or infections. It was a general practice to count the blood cells in medical diagnosis as it gives an overview of the general health of a particular patient. Haematologists used to manually count blood cells and do a microscopic examination, which was a time-consuming process. Moreover, the accuracy of counting is largely determined by their level of experience and physical health[1-3]. Cell counter machines have recently been employed in haematology centres to do CBCs, resulting in a rapid and reliable result. Due to its inability to distinguish cell shapes, certain irregular blood cells may not be identified accurately. Additionally, morphological abnormalities of cells cannot be detected by a cell counter in some disorders, so, a manual evaluation of the blood sample is required. As a result, multiple image processing techniques are utilised to segment and count WBCs and RBCs microscopically small blood samples[1].

When the MI as images of blood cell are collected from a microscope, there may be certain distortions and issues with illumination. These disturbances are defined as noise. Noise in blood cell images not only degrades the effectiveness of computerized diagnosis equipment but makes manual examination difficult. A challenge for researchers is how to denoise images of blood cells in medical imaging. This process involves denoising the images using different techniques that result in the estimation of the desired clear image. Several works of literature are reported for denoising. Anisotropic diffusion-based approaches[1-3] wavelet-based techniques[4], and nonlocal means-based techniques[5] have been mentioned as three important works carried out in the past for improving image quality. Filters with edge-stopping functions are used in the anisotropic diffusion-based technique[1]. It normally removes the noise from the images in such a way that the image edges are no longer blurred. A novel anisotropic diffusion-based filter is proposed[1,2], with the authors ensuring that noise estimation is a vital aspect of this technique. The authors suggested a bilateral filter for MRI denoising that works on the un decimated wavelet transform domain[3]. They have also used neighbourhood similarity to keep the edges of the images preserved. In the last decades' several denoising filters, based on nonlocal means-based techniques are reported in the literatures[4-6]. For the denoising process, these filters take advantage of data redundancy in the images. Nonlocal Means based filters, in reality, remove noise by averaging non-local

portions. The above-discussed three methodologies are, in fact, time-consuming and complex, necessitating the use of strategies or optimization to fine-tune the parameters. Recent advances in deep learning have demonstrated that a convolutional neural network (CNN) can analyze images more accurately on a wide range of tasks. CNN has succeeded at a variety of low-level image processing applications, including image super-resolution, image denoising, and visual dehazing, among others. This CNN technique has been extensively reported for blood cell images in the literature. The authors discovered a reasonable accuracy rate for detecting distinct WBC types, however, they did not include Basophils because there were not enough samples in the dataset[7]. The study demonstrates an approach to find Malaria using a deep belief network[8]; however, CNN's perform better when it comes to image classification. Zhang et al. have created a deep CNN model for additive Gaussian noise denoising[9]. They demonstrated that deep CNNs outperform classical approaches in terms of performance and time complexity. Furthermore, the suggested Dn-CNN, which uses a deep network structure and a residual learning technique to achieve an outstanding denoising effect, is beneficial in enhancing the capacity and flexibility for exploiting image characteristics. Unfortunately, noise is not something that can be learned. Not only does the learned residual image contain noise, but it also contains image detail, resulting in a loss of detail in the denoised image.

The authors developed deep feed forward denoising convolutional neural network for medical image denoising[10]. They studied the model in depth using a deep framework, a residual learning technique, and a regularisation strategy. This approach shows significant improvement in performance when compared with various other approaches. Jiang et al. developed a ten-layer CNN for MRI denoising in recent times[11]. For noise reduction, they used an architecture with very small (3x3) convolution filters. Pushing the depth[16-19], weight layers result in a significant improvement over previously reported configurations. In this paper, we have proposed a new CNN-based Blood cell image denoising approach, inspired by recent advances in CNN. The following are the main contributions of the proposed work:

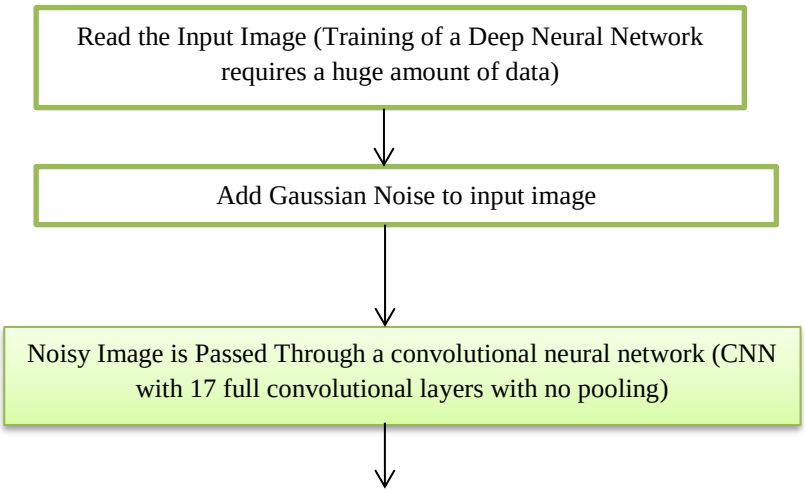
- A new CNN architecture is presented for denoising blood cell images. In the proposed architecture, the encoder-decoder structure is employed..
- The enhanced rectified linear units (ReLU)-CNNs are proposed for blood cell count diagnosis. The proposed method can improve from typical ReLU-CNN, and these benefits can help overcome the difficulties associated with vanishing gradients in conventional activation functions.

- The proposed method makes use of the residual learning strategy. The presented denoising framework employs both local and global residual learning. Local residual learning aids in the training process, while global residual learning aids in the preservation of significant image features.
- Simulated images of blood cell were used to create the training dataset. In the training images, we added synthetic Gaussian noise. The network was put to the test on both synthetic and real blood cell images.
- Weight optimization techniques are combined and investigated in order to obtain a better outcome with a zero Mean Square Error (MSE). The algorithms which are combined as Marine Predators Algorithm (MPA) and Tunicate Swarm Algorithm (TSA)

The paper is organised as follows. In Section 2, extensive literature survey is summarized; section 3 denoising the blood cell image using CNN architecture is discussed. Section 4 presents comparative evaluation of various hybrid algorithm is used to optimize the weight of CNN architecture .Section 4 gives the result analysis along with the dataset description utilized or denoising the images in terms of MSE, PSNR BCCD datasets . Section 5 concludes the paper

2. PROPOSED MEDICAL IMAGE DENOISING NETWORK

The proposed system intends to train an end-to-end mapping function in between the source image and residual image from its noisier counterpart. In terms of the design process, fundamental components, prominent features, and loss function, the subsequent sections provide a detailed explanation of CNN-MI. The fig.1 represents the complete work flow of denoising the MI images.



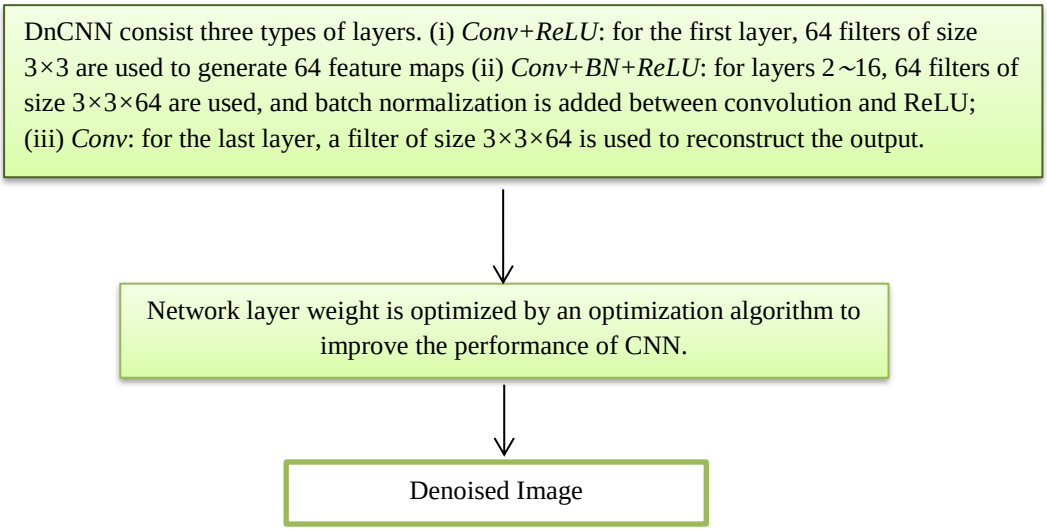


Fig 1 Flow chart of denoising MI using optimized CNN

a. Network architecture

The CNN is mainly composed of an input node, an output node, and several hidden nodes of neurons that conduct linear and non-linear modifications in alternating sequences is shown in the schematic block diagram of fig.2. A feature map is the output of each hidden node. Each node convolution kernel is multiplied by the previous node output. The activation functions process the input values after it has been multiplied by the convolution kernel weights. The methodology allows introducing Gaussian noise to the input image and then denoising through a CNN to improve BCI information accuracy. With a scale factor of 2, the network conducts down-sampling and up-sampling. Moreover, in this work, the denoising level is composed of 17 layers of depth, the first layer, which comprises Convolution, and the rectified Linear Unit which is mostly concerned with low information levels. Each convolution layer has 64 kernels in size of 3x3x1 filters, which are utilized to create 64 featured maps and the architecture includes a two-step down sampling process, which is followed by initial CONV+ReLU layers. These down-sampled features are fed to the residual block, in which the 2–16 layers include 64 filters of size 3x3x64 to create 64 image features, which comprise of convolution accompanied by ReLU and Batch Normalisation (BN). Local residual learning is included in each residual block and the purpose of these blocks is to obtain deep features required for denoising. To interpret the result of the preceding layers, the CNN performs two de-convolutions that up-sample the outcome of the residual blocks to 256x256x64 tensor. The up-sampled or decode output is given to the final layer that involves only the convolution layer of a 3x3x64 size filter. It is used to effectively advance and reconstruct the output image. Both local and global residual learning include the skip

connection. The four different types of residual blocks are used to accomplish local residual learning. In deep learning neural networks, an activation function is responsible for scaling operation and it changes the input to keep its values within a reasonable range. In general, the input node in CNN does not require any activation functions because the input layers values are scaled effectively[22].

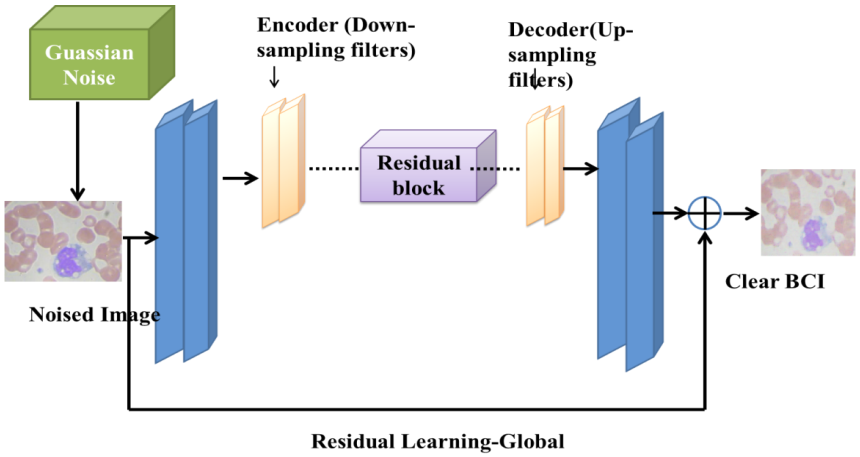


Fig 2 CNN architecture of denoising medical images

When the input parameters are multiplied by the value of weights and added to the very first hidden units, they quickly exceed the range of their initial scale, necessitating the use of activation functions. These huge numbers can be pushed back into the acceptable range by activation functions, making them usable.

The vanishing gradient in CNN is a problem that occurs when using step size to train neural network models. When more layers with specific activation functions are added to neural networks, the gradients vanish, making the network difficult to train. In order to avoid gradient vanishing issues, the proposed work modifies the activation function of the ReLU layer in CNN namely, Leaky Rectified Linear Unit (LReLU) and parametric Rectified Linear Unit (PReLU). Unlike traditional activation functions like sigmoid, hyperbolic tangential, the improved ReLU function and its variants do not require complicated numerical computation (converges between 0 and 1), hence the computing speed is improved. The activation functions of LReLU and PReLU is represented as

$$f_L(x) = \begin{cases} x & x > 1 \\ 0.01x & otherwise \end{cases} \quad (1)$$

$$f(x) = \max(0.01x, x) \quad (2)$$

$$f_P(x) = \begin{cases} x & x > 1 \\ px & otherwise \end{cases} \quad (3)$$

$$f(x) = \max(px, x) \quad (4)$$

Where $f_L(x)$ and $f_P(x)$ is the activation function of Leaky ReLU and Parametric ReLU respectively, x is the input value, instead of fixed slope value of LReLU $(0.01x)$ use "p" as parametric slope value (x<0) in PReLU. Both activation functions can be viewed as an expanded versions of the ReLU function. To investigate the impact of enhanced ReLU-CNNs, LReLU-CNN and PReLU-CNN are employed separately. Furthermore, when p equals 0.01, LReLU-CNN is a PReLU expert.

Batch normalisation speeds up and improves the accuracy of network training. Normalizing each layer input may occasionally affect what the layer is supposed to represent and which is solved through identity transform[23]. In general BN is trained along with data computation and can be represented as

$$\widetilde{x_N} = \frac{x_N - E(x_N)}{\sqrt{Var(x_N)}} \quad (5)$$

In order to optimize the aforementioned problem, transformation pair parameter γ_N and η_N for each x_N is inserted for shifting and scaling the input batch images and it is defined as,

$$y_N = \gamma_N x_N + \eta_N \quad (6)$$

Where x_N is the normalised input image, $E(x_N)$ is the expectation value of x_N, $Var(x_N)$ is the variant of x_N, γ_N and η_N. x_N, γ_N and η_N respectively, are transformation parameters and are trained along with variable in the model. In addition to that the weight ratio is optimized by recent optimized techniques like TSA-Tunicate Swarm Algorithm, MRFO-Manta Ray Foraging Optimization algorithm, AOA-Arithmetic Optimization Algorithm, JS-Jelly Fish optimization algorithm, SOA-Seagull Optimization Algorithm and the results are compared with proposed hybrid algorithm .

b Encoder – decoder

An encoder and a decoder are part of the De-CNN structure. It is made up of three parts: encoding, residual unit, and decoding. The encoding downsizes the input vector after performing various changes on it. These layers encode a 256 x256x 64 to 64x 64x128 tensor input. The residual units extract deep characteristics that can be used for denoising. Decoding is a process of up-sampling to recover geographical information whereas down sampling is used to capture semantic or context information. This method is particularly useful for preserving important visual features while ignoring worthless. As a result, the network has a higher generalization capability than a traditional CNN.

c Residual learning

The term "residual learning" refers to the creation of a short path between the input image and the output block. The residual mapping will be much easier to optimize if the initial

mapping is the same as an identity mapping. The developed denoising network makes use of residual learning for faster and more effective training. Local residual learning is implemented in the middle of the network or immediately after the down sampling phase. Four residual blocks are used in the network's centre, which performs local residual learning. The skip-connection is present in every residual block which has 128 filters and is made up of Conv+BN+LReLU. The proposed network additionally takes into account global residual learning and a skip-connection is particularly introduced. For employing global residual learning, is illustrated in fig 1. As a result, the architecture functions as only one global residual block exists. This type of approach assists the network in preventing any unexpected data loss during the intermediate phases of the denoising process. The residual learning technique, which employs a CNN network, increases model precision substantially. As a result, we explore the advantages of residual learning for convolutional neural networks and use residual learning to medical picture denoising.

d Loss Minimization

According to CNN Architecture Network, during the traditional denoising procedure, some picture detail information is lost.B stands for the picture detail message that was lost at the time of denoising process. The intermediate feature map is characterized as the prime denoised picture generated through the conventional denoising procedure[24]. The learning phenomenon is to reduce the distinction between B and A if the image required data retrieved by an elaborateness preserving review procedure is represent as A. The mean square error may be used to create the loss function:

$$Mean\ Squared\ Error\ (MSE) = \frac{1}{R \times S} \sum_{i=1}^{R} \sum_{j=1}^{S} \left\| A(i,j) - \theta\big(B(i,j)\big) \right\|_2^2 \qquad (7)$$

Where θ signifies a training data or set of parameters that may be used to create the estimated BCI image. It should be observed that A and B are RxS in size.

3. WEIGHT OPTIMIZING ALGORITHM

SOA + MPA Algorithm

In case1 SOA and MPA are combined to denoise the blood cell images. MPA splits the iterations to perform distinct functions as a global optimization technique. The one third of the iteration is dedicated to global exploration, the second third to half of the particle exploration and half of the particle exploitation, and the last third to partial development[27-29]. It is evident from tables 1 and 2 the performances are improved. For noise variation of 0.1 shown in table 1, the MSE, MAE and SMAPE, have significantly reduced when compared with the values obtained for 0.2 noise variation. At the same time there are no considerable

change in the NC and SSIM. But there is a significant reduction in PSNR and PC for 0.2 noise variation in comparison with 0.1. PSNR is the indication for the image quality and hence it is understood that there is an increase in the quality of the denoised images. But the major drawback in this combinational algorithm is that the time consumption is comparatively more than all the other combinations, as illustrated in fig.7. The MPA optimization algorithm includes the following scenario:

Stage 1: This is the beginning or exploratory phase. The assumption here is premised on the reality that the predator goes quicker than the prey. This phase's mathematical model is specified as follows:

$$\left.\begin{aligned} &while\ Iteration < \frac{1}{3} Maxi\_Iteration \\ &\overrightarrow{stepsize_J} = \overrightarrow{K_B} \times \left(\overrightarrow{Elite_J} - \overrightarrow{K_B} \times \overrightarrow{Prey_J}\right), j = 1,2 \cdots\cdots\cdots n \\ &\overrightarrow{Prey_J} = \overrightarrow{Prey_J} + Q.\overrightarrow{K} \times \overrightarrow{stepsize_J} \end{aligned}\right\} \quad (13)$$

Stage 2; This is an intermediate phase wherein the exploration is attempting to be transiently transformed to exploitation. Exploration and exploitation are significant in this phase. As a result, 50 % of the population is allocated for exploration, while the remaining half is allocated for exploitation. During this phase, the prey is in charge of exploitation and the predator is in charge of exploration. According to the rule, if the prey moves in Levy at a unit velocity ratio (v ≈1), the ideal predator approach is Brownian. As a result, this research addresses prey movements in Levy and predator movements in Brownian and is defines as,

$$while \frac{1}{3}\ Maxi\_Itereration < Iteration < \frac{2}{3} Maxi\_Iteration$$

In the case of the first 50 % of the population,

$$\overrightarrow{stepsize_J} = \overrightarrow{K_L} \times \left(\overrightarrow{Elite_J} - \overrightarrow{K_L} \times \overrightarrow{Prey_J}\right), j = 1,2 \cdots\cdots\cdots n/2$$

$$\overrightarrow{Prey_J} = \overrightarrow{Prey_J} + Q.\overrightarrow{K} \times \overrightarrow{stepsize_\iota} \quad (14)$$

where $\overrightarrow{K_L}$ is a vector of random integers based on the Lévy distribution that represents Lévy motion. The study implies for the later part of populations:

$$\overrightarrow{stepsize_J} = \overrightarrow{K_B} \times \left(K_B \times \overrightarrow{Elite_J} - \overrightarrow{Prey_J}\right), j = n/2 \cdots\cdots\cdots n$$

$$\overrightarrow{Prey_J} = \overrightarrow{Elite_J} + Q.CF \times \overrightarrow{stepsize_J}$$

While $CF = \left(1 - \frac{Iteration}{Maxi\_Iteration}\right)^{\left(2\frac{Iteration}{Maxi\_Iteration}\right)}$ is regarded as an adaptive variable that regulates the step length for predator mobility. Multiplication of $\overrightarrow{K_B}$ and Elite represents

predator movement in Brownian motion, whereas prey changes its location dependent on predator movement in Brownian motion.

Stage 3: The final stage of optimization and is denoted by,,

$$\left.\begin{array}{l} while\ Iteration > \frac{2}{3}\ Maxi\_Iteration \\ \overrightarrow{stepsize_J} = \overrightarrow{K_L} \times \left(K_L \times \overrightarrow{Elite_J} - \overrightarrow{Prey_J}\right), j = 1 \cdots \cdots \cdots n \\ \overrightarrow{Prey_J} = \overrightarrow{Elite_J} + Q.CF \times \overrightarrow{stepsize_J} \end{array}\right\} \quad (15)$$

In the Lévy method, multiplying $\overrightarrow{K_L}$ and Elite mimics predator movement, whereas applying the step size to Elite position represents predator movement to aid in the updating of prey location.

Now, utilising Eq (A), the head and follower selection techniques are used to update the locations of search agents. In order to get the required outcome, the SOA exploitation phase is included. The SAO exploitation phase is defined as follows:

The exploitation attempts to capitalise on the search process's knowledge and experience. While migration, seagulls may alter their assault angle as well as their pace. They keep their altitude up by utilising their wings and bodyweight. The spiral motion behaviour happens mostly in air when attacking the prey.

$$\begin{array}{l} a' = rcos(l) \\ b' = rsin(l) \\ c' = r \times l \\ r = p \times e^{kq} \end{array} \quad (16)$$

where r is distance of each spiral turn and k is a random integer in the range $[0 \leq l \leq 2\pi]$. p and q are constants that define the spiral form, while e is the natural logarithm base. Using the aforementioned equations, the updated location of the search agent is defined as

$$\overrightarrow{T_s}(x) = \left(\overrightarrow{L_s} \times a' \times b' \times c\right) + \overrightarrow{T_{bs}}(x) \quad (A)$$

where $\overrightarrow{T_s}(x)$saves the best solution and updates the position of other search agents

The pseudocode for the hybrid combination of MPA+ SOA is as follows:

While the termination requirements are not satisfied, initialise searching agent (Prey) populations j=1,...,n. Evaluate fitness and create the Elite matrix.

if Iteration<Maxi_Iteration/3

Restore prey value using Eq.(13)

else if Maxi_Iteration/3<Iteration<2*Maxi_Iteration/3

The initial half of populations (j=1,…,n/2)

Restore prey value using Eq.(14)

For the other half of the populations (j=n/2,…,n)

Restore prey value using Eq.(15)

else if Iteration>2*Maxi_Iterationr/3

Restore prey value using Eq.(16)

end (if)

While (j, $i < maxi$)

for $j = 1$ to Pop

update

end for

for every search agent

update current search agent location

else

update SOA position using Eq. (A)

end for

update best solution

$i = i + 1$

End while

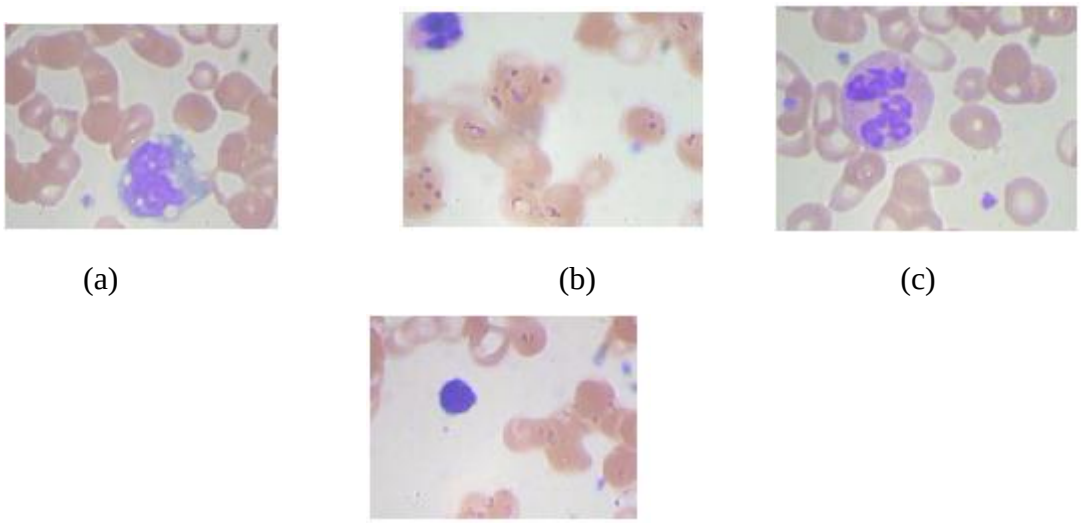
End while

Return

4. RESULTS AND DISCUSSION

Descriptions of the dataset and the network training

The BCCD dataset (https://github.com/Shenggan/BCCD Dataset) and a publically available dataset (https://www.kaggle.c-om/paultimothymooney/blood-cells/data) were used to obtain our raw data. A recent dataset containing 12,444 augmented BCI in JPEG format was obtained, with 9,957 and 2,487 training and validation sets images, respectively. Blood cells are classified by their kind and it is categorized these images into four types. Eosinophils, lymphocytes, monocytes, and neutrophils are the different cell types. There are 2,497 eosinophils, 2,483 lymphocytes, 2,478 monocytes, and 2,499 neutrophil images in the training dataset, and 623 eosinophils, 623 lymphocytes, 620 monocytes, and 624 neutrophil images in the test dataset. The full network used RGB images at a resolution of 320x240x3 pixels[25]. The various types of cell images are shown in Figure.3



(a) (b) (c)

(d)

Fig 3 Dataset including 12,444 cell images of (a) Monocyte (b) Eosinophil (c) Neutrophil (d) Lymphocyte

The blood cell images obtained from the datasets were divided into three categories: training, validation, and testing. The training set consists of 65 images, while validation and testing sets each contain 15 images. We have artificially inserted Gaussian noise in the range of these sets. The training, validation and test sets have 1950,450 and 450 blood cell images, respectively, after adding noise. The proposed denoising network has been trained end-to-end on this dataset. And the denoise performance is measured by using the following functions:

(i) $Mean\ Squared\ Error\ (MSE)$ is clearly represented in loss minimization section by using equation (7)

(ii) $Peak\ Signal\ Noise\ Ratio\ (PSNR) =$

$$10log_{10}\left(\frac{(max(A))^2}{MSE}\right) Normalized\ cross-corellation\ (NC) = \frac{\sum_{i=0}^{M-1}\sum_{j=0}^{N-1} AB}{\sum_{i=0}^{M-1}\sum_{j=0}^{N-1} A^2} \quad (8)$$

Where A is the actual image, B is represented as details lost image during denoising process and F is the forecasted value. The θ signifies a training data or set of parameters that may be used to create the estimated BCI image. It should be observed that A and B are $R \times S$ in size. Both theoretical and practical results have been used to investigate the performance parameters of the proposed novel method. For quantitative analysis, various performance measures such as MSE and PSNR are used. The Gaussian noise is added with a noise ratio of 0.1 initially and the noised image is shown in figure.4. Figure 5 shows the qualitative denoising performance for various blood cell images using SOA-MPA algorithms with noise variation of 0.1.

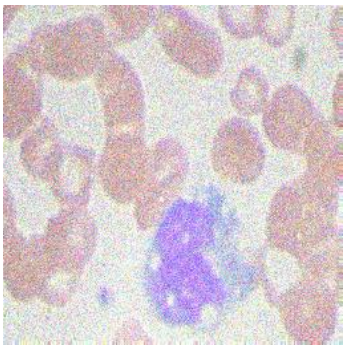


Fig 4 Noised Image with 0.1 noise variation

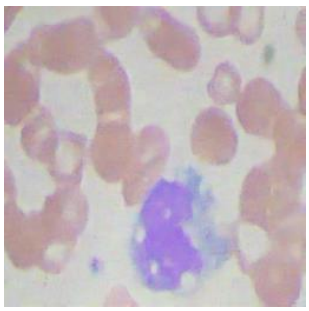


Fig 5 Denoised blood cell images using MPA-SOA+DnCNN algorithm for 0.1 noise variation.

The results thus obtained for denoising the images are extremely comparable to the clear MI, and distortions are not at all visible in the resultant images without any noise as can be seen from these results. The structure of the blood cell images in the denoised image is highly retained, as can be seen in figures 5 and 6. This shows that the various approaches utilized not only efficiently decrease noise but also preserve structural detail in blood cell images. The quantitative results for both the case are illustrated in Tables 1 and 2 respectively. On comparing these two tables it is well understood that the performance is improved when the noise variation is increased from 0.1 to 0.2. These results suggest that the network developed with high levels of noise and evaluated with low levels of noise performs better than the system trained with almost the same levels of noise. In this paper Seagull optimization algorithm and MPA are combined together to achieve the desired result, since, SOA is one of the simplest algorithms with more benefits. SOA does not get caught in local optima, instead searches the entire domain. The sample point distribution is centred on the real optimal solution, ensuring that it can be exploited. As a result, SOA has the potential to explore as well as exploit. It is highly accurate as the solution gets approximated while the simulation is still running. SOA has a high computational success rate when it comes to solving optimization problems. But major drawback lies in the fact that in the initial phases of

optimization, the algorithm exhibits massive and sudden changes [19]. This in turn enables to combine the SOA with algorithm MPA.

Table 1 MPA-SOA algorithm for 0.1 and 0.2 noise variation

Technique	Noise variation	PSNR	MSE
MPA-SOA	**0.1**	32.5007	0.00056226
DnCNN	**0.2**	22.6888	0.0053842

Table 2 Experimental results MPA-SOA algorithms for 0.1 and 0.2 noise variation

Algorithm	Noise variation	Best	Worst	STD	Mean
MPA-SOA	**0.1**	5.662603e-04	6.589569e-04	2.481290e-05	5.735293e-04
	0.2	5.384200e-03	7.843533e-03	5.486863e-04	5.562281e-03

Tables 1 and 2 illustrates the experimental data. For 0.1 noise variation the best value is obtained as 5.662603e-04 for the hybrid combination of MPA and SOA

5. CONCLUSION

For the removal of noise in MI images, the paper presented CNN-MI, a novel deep learning CNN approach. Multiple convolutions are used in the proposed CNN to capture distinct image characteristics while isolating inherent noise. Furthermore, the suggested technique uses an encoder-decoder structure to conduct image sampling both in the downward and upward methods during the process of removing the noise from the BCI. The suggested CNN model additionally incorporates residual learning and simulated blood cell images are used for network training. To obtain better performance, the improved ReLU-CNNs employs LReLU and PReLu activation function, that yields more benefits than the conventional ReLU. The enhancements on datasets, ReLU-CNNs can achieve a better diagnosis level of accuracy and exceed the classical ReLU-CNN in terms of convergence. Several tests on real MI images show that the suggested approach delivers promising theoretical and practical denoising results. The noise is successfully eliminated, and the mean square error (MSE) is lowered to zero, owing to extensive modelling. Furthermore, the network's performance has

been evaluated using modified weight optimization strategies, and the proposed optimized Dn-CNN outperforms existing methods.

REFERENCE

(1) G. Gerig , O. Kubler , R. Kikinis , F.A. Jolesz , Nonlinear anisotropic filtering of MRI data, IEEE Trans. Med. Imaging 11 (2) (1992) 221–232 .

(2) K. Krissian , S. Aja-Fernández , Noise-driven anisotropic diffusion filtering of MRI, IEEE Trans. Image Process. 18 (10) (2009) 2265–2274 .

(3) J. Tang , Q. Sun , J. Liu , Y. Cao ,An adaptive anisotropic diffusion filter for noise reduction in MR images, in: International Conference on Mechatronics and Automation, 2007, pp. 1299–1304 .

(4) C.S. Anand , J. Sahambi , MRI denoising using bilateral filter in redundant wavelet domain, in: IEEE TENCON, 2008, pp. 1–6 .

(5) A. Buades , B. Coll , J.-M. Morel , A non-local algorithm for image denoising, in: IEEE Conference on Computer Vision and Pattern Recognition, 2, 2005, pp. 60–65 .

(6) G. Chen , P. Zhang , Y. Wu , D. Shen , P.-T. Yap , Denoising magnetic reso- nance images using collaborative non-local means, Neurocomputing 177 (2016) 215–227 .

(7) J.V. Manjón , J. Carbonell-Caballero , J.J. Lull , G. García-Martí, L. Martí-Bonmatí, M. Robles , MRI Denoising using non-local means, Med. Image Anal. 12 (4) (2008) 514–523.

(8) N. Theera-Umpon and P. D. Gader, ``System-level training of neural networks for counting white blood cells," IEEE Trans. Syst., Man, Cybern. C,Appl. Rev., 32(1) (2002), pp. 48-53, .

(9) D. Bibin, M. S. Nair, and P. Punitha, “Malaria parasite detection from peripheral blood smear images using deep belief networks,” IEEEAccess, vol. 5, pp. 9099–9108, 2017.

(10) Starck JL, Cands EJ, Donoho DL, “The curvelet transform for image denoising”, IEEE Tran Image Process 11(6) (2002) 670–684

(11) D. Jiang , W. Dou , L. Vosters , X. Xu , Y. Sun , T. Tan , Denoising of 3D magnetic resonance images with multi-channel residual learning of convolutional neural network, Jpn. J. Radiol. 36 (9) (2018) 566–574 .

(12) Zhang K, ZuoW, Chen Y, Meng D, Zhang L, “Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising”, Tech report Computer Vision and Pattern Recognition, 2016, pp 1–13.

(13) Acharya V, Kumar P. Identification and red blood cell automated counting from blood

smear images using computer-aided system. Med Biol Eng Comput. 2018 vol 56 no 3:pp-483-489.

(14) Acevedo A, Alférez S, Merino A, Puigví L, Rodellar J. Recognition of peripheral blood cell images using convolutional neural networks. Comput Methods Programs Biomed. 2019,180:105020.

(15) Almezhghwi, Khaled, and Sertan Serte. "Improved Classification of White Blood Cells with the Generative Adversarial Network and Deep Convolutional Neural Network." Computational intelligence and neuroscience vol. 2020 6490479. 9 Jul. 2020.

(16) H. Kutlu, E. Avci, and F. Özyurt, ''White blood cells detection and classification based on regional convolutional neural networks,'' Med. Hypotheses, vol. 135, Feb. 2020.

(17) F. Özyurt, ''A fused CNN model for WBC detection with MRMR feature selection and extreme learning machine,'' Soft Comput., vol. 24, pp. 8163–8172, Jun. 2019.

(18) Xufeng Yao, Kai Sun, Xixi Bu, Congyi Zhao & Yu Jin Classification of white blood cells using weighted optimized deformable convolutional neural networks, Artificial Cells, Nanomedicine, and Biotechnology, (2021) vol 49:no.1, pp 147-155.

(19) Liang, Gaobo & Hong, Huichao & Xie, Weifang & Zheng, Lixin.. Combining Convolutional Neural Network With Recursive Neural Network for Blood Cell Image Classification. IEEE Access. (2018) PP. 1-1

(20) Sahlol, A.T., Kollmannsberger, P. & Ewees, A.A. Efficient Classification of White Blood Cell Leukemia with Improved Swarm Optimization of Deep Features. Sci Rep (2020) vol 10 no.1

(21) V. Jeya Ramya, Dr. S. Lakshmi, "Enhanced Deep CNN based Arithmetic Optimization Algorithm for Acute Myelogenous Leukemia Detection", Annals of RSCB, 2021,pp. 2999.

(22) A.M. Patil, M.D. Patil, G.K. Birajdar,White Blood Cells Image Classification Using Deep Learning with Canonical Correlation Analysis,IRBM,2020

(23) Sarah Ali Abdelaziz Ismael, Ammar Mohammed, Hesham Hefny, "An enhanced deep learning approach for brain cancer MRI images classification using residual networks", Artificial Intelligence in Medicine, Volume 102,2020.

(24) J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, "Deep learning for smart manufacturing: Methods and applications", J. Manuf. Syst.,(2018) 48(1) 144-156.

(25) A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks", in Proc. Int. Conf. Neural Inf.Process. Syst., 2012 60(1),1097-1105.

(26) A. Kaveh, S. Talatahari, A novel heuristic optimization method: charged system search, Acta Mech,2010 213 (3–4) 267–289.

(27) H. Liu, Z. Cai, Y. Wang, Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization, Appl. Soft Comput. 10 (2), 2010,pp-629–640.

(28) I.Ahmadianfar,O. Bozorg-Haddad, and X. Chu, “Gradient-based optimizer:A new metaheuristic optimization algorithm", Inf. Sci., 2020,540(1),pp. 131-159.

(29) Faramarzi A , Heidarinejad M , Mirjalili S ,et al, “Marine Predators Algorithm:A Nature-inspired Metaheuristic”,Expert Systems withApplications,2020, 152,1-49.

Received 11 Dec 2020
Revised 22 Jan 2021
Accepted 4 Feb 2021
Published Online 27 Feb 2021