Malaria Disease Prediction Based on Machine Learning

Octave Iradukunda[1], Haiying Che[1*], Josiane Uwineza[1],
Jean Yves Bayingana[1], Muhammad S Bin-Imam[1], Ibrahim Niyonzima[1]

[1]School of Computer Science and Technology, Beijing Institute of Technology, Beijing, China
[2]School of Automation, Beijing Institute of Technology, Beijing, China
Email address: liloctave@gmail.com, chrissieche@qq.com, inezajoyorange@gmail.com

Malaria detection is a stressful job for most doctors and it requires experiences and expertise. The machine learing (ML) method can be used to releave this issue. This paper try to find suitable model to help detect malaria with accuracy. The used dataset was released by National Institute of Health in USA and contained a total number of 27,560 red blood cell (RBC) images with equivalent instances of parasitized and uninfected RBCs images. A single hidden layer feedforward neural networks methodology namely Extreme Learning Machine (ELM) model was applied to classify and predict whether a patient has been affected by malaria or not. ELM has been compared with other machine learning techniques like SVM, KNN, CART, RF, CNN, VGG16, RESNET, and DENSENET, and it has outperformed all the other with 99% of accuracy, $28seconds$ cost time, 0.0095 Misclassification Error, and 98% precision which showed the effectiveness of ELM in the application of malaria cell detection scenario and it can also be referred by other researchers in the related field.

***Index Terms*—Malaria Red Blood Cells, Extreme Learning Machine (ELM), DenseNet, RESNET, VGG16, Machine Learning (ML), and CNN-Keras**

I. Introduction

Malaria is a blood disease which has become enormous problem all over the world [1]. Its causes numerous signs that characteristically fever, fatigue, vomiting, and headaches. If not treated swiftly and appropriately on time, it can cause yellow skin, diarrhea, profuse sweating, muscle pain and pain in joints, seizures, coma, and leading to death.

Microscope is the common method used by the microscopists to identify malaria parasites in human red blood cells. But microscope is not standardized itself to predict if the patient's RBCs is infected by malaria or not as it requires the ability, experience, and expertise of the microscopist to detect it by interpreting the results given by the microscope [2] which is not reliable in case that the microscopist is a non-expert [3].Low accuracy of malaria detection, consequently lead the doctors to give the false drugs to the patients where the RBCs continue to lose their immunities which can lead the patients to death in most cases or sometimes to be re-examine as the patient's illness continue to progress on high level than to be improved.

Identify applicable funding agency here. If none, delete this.

A. Related works

To quickly diagnose malaria several algorithms were proposed. Generally, as stated by Tek et al. [4], almost all work done for malaria RBCs diagnosis have followed these steps:

1) To acquire the digital images of RBCs;
2) Image preprocessing to improve quality of image and reduce variation of image by removing noise, normalize lighting and variation of color for the acquired images, and staining process to automate analysis of digital RBCs [5] [6];
3) Detection and segmentation of distinct RBCs [7];
4) Features extraction and selection [8], and
5) Identification and labeling to classifies the t feature vectors into distinctive classes where each RBCs is labeled as either infected or uninfected.

Liang et al. [9], are the first ones who published a paper by applying Machine Learning (ML) to malaria diagnosis where they used a CNN to discriminate between infected and uninfected cells in thin blood smears after using a conventional level-set cell segmentation approach. One of the biggest advantages of using ML is that it doesn't require the design of hand-crafted features and segmented RBCs are usual input for a CNN. Dong et al. [10] used CNNs after applying ML to cell segmentation, and lately Hung et al. [11] presented end-to-end framework using faster Region-based CNN. However, CNN typically requires large training sets and time.

The most classification techniques applied in previous works for classification of infected and uninfected malaria RBCs include K-nearest neighbors' classifier (KNN) [12], Support Vector Machine (SVM) [13], Artificial Neural Network (ANN) [14], Naive Bayes [15], Feed Forward neural network [16], etc.

B. Goal and Contribution

The goal of this paper is to find a trained model that can predict malaria disease with high accuracy. This will enable doctors with more precise diagnosis. The main contribution of this paper is to evaluate the performance of ELM with other machine learning methods on the same datasets

This paper is schematizing in distinctive sections as follows: Section 1 is the 'Introduction '; Section 2 is 'Applied Method'; Section 3 is 'Experiment, Results and Discussion'; and finally Section 4 is 'Conclusion and Future work'.

978-1-7281-2345-5/19/$31.00 ©2019 IEEE

II. Applied Methodology

A. Preprocessing

The preprocessing was done to reduce the noises and variations, and standardize the RBCs images' size. Fig.1. shows how the RBCs images was processed in this paper.

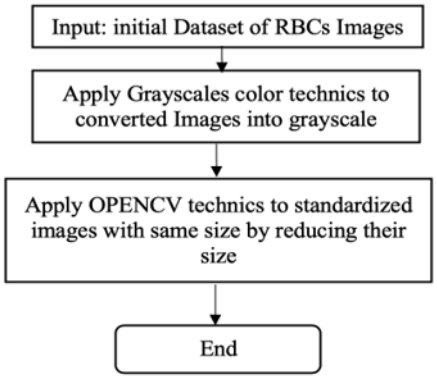

Fig. 1. Preprocessing flowchart.

The first step was to convert RBCs' images into grayscale using grayscale color technique to improve the quality of images. The second step was to reduce the size of RBCs images, where the images were re-sampled to 500 × 500-pixel resolutions to satisfy the input restriction during the training process. In the meantime, RBCs images were standardized to support quicker convergence. The both processes will make the images suitable for further treatment.

B. Features extraction

The feature extraction was done to convert the RBCs images into feature vectors. The combination of 3 different feature extraction methods was applied in order to get the appropriate features from RBCs images which are invariant to the scale, translation, and rotation as better features leads to get higher recognition accuracy. The combined methods are Hu-Moments for extracting features according to the shape of images as shown by Zhihu Huang and Jinsong Leng [17], Haralick-Texture [18] for extracting features according to the images texture and Color Histogram for extracting features according to the color of images. The reason of choosing these 3 methods among many is that they are the ones to calculate image feature vectors from 3 different characteristics of image. Fig.2. shows how the features were extracted from the RBCs images.

The first method is Hu-moment which was applied to preprocessed RBCs images. It extracts features based on their shapes and resulted features are invariant to translation, rotation and scaling. Also, the fluctuation of RBCS images was decreased as their spatial resolution of images increased; and computation increased quickly the resolution of RBCs images. The Hu moment features was obtained by combining

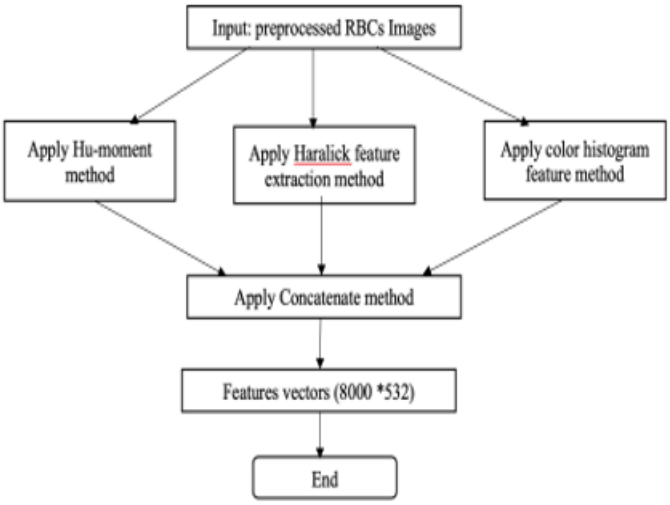

Fig. 2. Features Extraction Flowchart.

7 different order RBCs images moments and the mathematical model of image moment is:

$$\mu_{m,n} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x - x_c)^m (y - y_c)^n f(x, y) \qquad (1)$$

The second applied method is Haralick feature extraction method to extract textual features from preprocessed RBCs images. This feature extraction method was applied into two steps in this paper:

1) To compute the Gray Level Co-occurrence Matrix (GLCM) which defined how the pixels in the binary of RBCs images are spatial related, and
2) To compute texture features of these binary images using 13 equations of Gray Level Co-occurrence Matrix which are known as Haralick texture features

The third method was to apply color histogram feature extraction method to preprocessed RBCs images, this extracted features from RCCs images based on different color represented in image and display them in the form of histogram.

After applying those 3 features extraction methods to preprocessed RBCs images, concatenate method was used to combine the features from those methods and come up with the combined features vector of 8000×532 dimensions (numbers of RBCs images times number of extracted features) which used as input in the stage of classification.

C. Overview of ELM

The applied classification method in this paper is ELM which was recommended by Professor Huang [19] [20]. ELM is logically feedforward neural networks (FNN) and was developed from single hidden layer feedforward network (SLFN). Later, ELM was extended to a generalized SLFN [21]. The SLFN has been adjudged by researchers as having

the ability to learn effectively with tolerable errors using N hidden neurons and any activation function.

ELM is made by three layers which are input layer, hidden layer and output layer. The parameters of ELM which are input weights and biases are arbitrarily assigned and due to that ELM reaches the smallest norm of weights [22] [23]. The output weights between the hidden nodes and the output neuron(s) are analytically computed and learned in a single step. Therefore, there is no more dependency between the parameters (weights and biases) of the hidden layer as it was in the traditional feedforward networks that necessitated the tuning.

1) Difference of ELM compared with other Machine Learning

Different from BP, SVM and other ML algorithms which consider multi-layer of networks as a black box. While, ELM considers multi-hidden-layer of networks as a white box and trained layer-by-layer. ELM models have produced good generality performance and learn thousands of epochs quicker compare to networks trained through backpropagation (BP). However, different from other neural networks algorithms which require intensive tuning in hidden neurons, ELM theories show that hidden neurons are important, but do not demand to be tuned during the training.

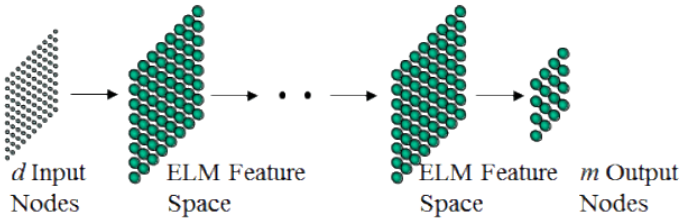

Fig. 3. ELM model.

The commendable attributes of ELM are simple learning algorithm, reach smallest norm weight and smaller training error [22], has good generalization performance and accuracy [33], improved efficiency [22], an unified solution to different practical applications [21], a faster speed and cost reduction [24], non-linear transformation in its training phase, devoid of local minimal and overfitting [21] [22], requires fewer optimization when compared with SVM, though it has the same computational cost with SVM.

To increase the capability of ELM models, the activation functions are used in hidden layer and those activation functions may be: sigmoid, tanh, hard limit, sine, gaussian, and multi-quadratic activation functions and so on [25].

2) ELM Algorithm

For standard single hidden layer feed forward neural network: Having N distinct arbitrary samples (x_i, t_i) where $x_i = [x_{i1}, x_{i2}, \dots x_{in}]^T \in R^n$ where x_i is the input and $t_i = [t_{i1}, t_{i2}, \dots t_{im}]^T \in R^m$ where t_i is the target and Having L hidden neurons and activation function g(x), [20]ELM is mathematically modeled as:

$\sum_{i=1}^{L} \beta_i g(w_i x_j + b_j) = o_j$

where: $w_i = [w_{i1}, w_{i2}, \dots w_{in}]^T$: it is input weight vector

$B_i = [\beta_{i1}, \beta_{i2}, \dots \beta_{in}]^T$: output weight

$o_j = \{o_{j1}, o_{j2}, \dots, o_{jn}\}$: output vector of the network

For standard SLFNs, $\exists\{\beta_i, w_i, b_j\}$ such that $\sum_{i=1}^{L} \|o_j - t_j\| = 0$ and in this case: $\sum_{i=1}^{L} \beta_i g(w_i x_j + b_j) = t_j$ and this can be simply written as $H\beta = T$

where :

$$H([w_1, w_2, ...w_L, b_1, b_2, ...b_L, x_1, x_2, ...x_N]) = \begin{bmatrix} g(w_1 x_1 + b_1) & \cdots & g(w_L x_j + b_L) \\ \vdots & \ddots & \vdots \\ g(w_1 x_N + b_1) & \cdots & g(w_L x_N + b_L) \end{bmatrix}_{N \times L}$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{bmatrix}_{N \times L} \text{ and } T = \begin{bmatrix} T_1^T \\ \vdots \\ T_N^T \end{bmatrix}_{N \times m}$$

Most of the case the number of input data N into ELM is not equal to the number of hidden neurons, this may result in ELM model being overdetermined (when number of input is greater than the number of hidden neurons $N > L$), being determined (when number of input is equal to the number of hidden neurons ($N = L$) or being underdetermined (when number of input is less than the number of hidden neurons ($N < L$) [26]. When the model is overdetermined and underdetermined H is not inverible. Then, to compute the output weights β and to solve that overdetermined problem, minimum norm was found by applying Moore-Penrose generalized inverse of matrix H and was denoted as $H^+ = (H^T H)^{-1} H^T$ and output weight matrix β was calculated as follows: $\beta = H^+ T = (H^T H)^{-1} H^T T$

Finally, the steps which involve in the ELM algorithm can be summarized as the follows: Given a training set $N = (x_i, t_i)$, $x_i \in R^n$, $t_i \in R^m$, $i = 1, \cdots, N$, activativation function $g(x)$, and hidden neuron number L:

Step 1: assign arbitrary input weights w_i and biases b_i

Step 2: calculate the hidden layer output matrix H

Step 3: calculate the output weight using β using $\beta = H^+ T$

Fig.4. shows how ELM was used in classification and prediction of RBCs images.

III. Experiment, Results, and Discussions

A. Hardware specification and Environments

The experiment was conducted in a GPU laptop PC with intel core i7 3.5GHz processor of 32GB RAM, NVIVIA and the operating system of Linux Ubuntu.The experiment was conducted under the environment with python version 2.7 and 3.6. Python idle and Jupyter Notebook were used to execute the codes. Open CV library together with python libraries were used; and Keras and Tensorflow were used as backend.

B. Description of Used Dataset

In this paper, the data was acquired from the Lister Hill National Center for Biomedical Communications (LHNCBC),

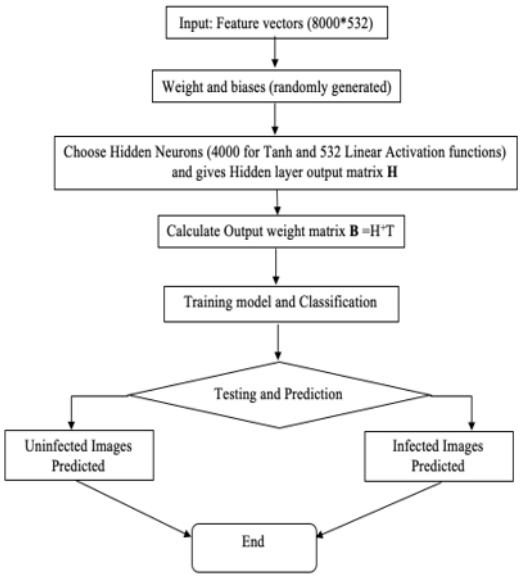

Fig. 4. Training and classification flowchart.

TABLE I
DESCRIPTION OF DATASET

Descriptions	*Parasites/Infected*	*Uninfected*	*Total*
Number of images	13780	13780	27560
Number of Patients	150	50	200
Format	PNG	PNG	-
Size of Dataset	-	-	407MB
Position of Images	V and H	V and H	-
Position of Images	RGB	RGB	-

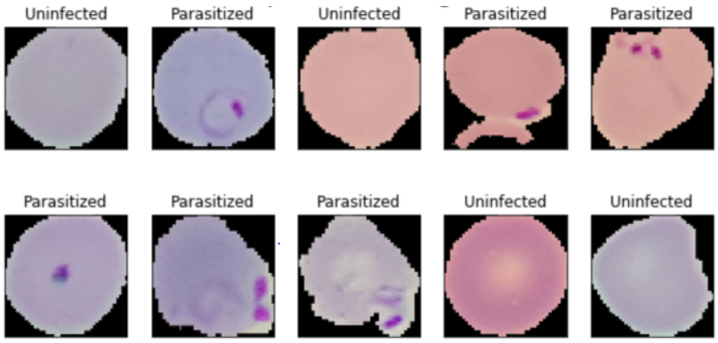

Fig. 5. Example of images to be used during the training and testing process.

part of National Library of Medicine (NLM), the dataset containing a total number of 27,560 red blood cell images with equivalent instances of parasitized and uninfected cells. The dataset is labeled where Each image is labeled as parasitized or uninfected. Moreover, it contained the CSV files holding the Patient-ID to cell mappings for the 2 classes of parasitized and uninfected, but the csv files were not used as in this paper images were studied separately not based on specific patient. The attributes of the dataset are patient-ID to cell mappings for the infected and uninfected label of image, and some images are in vertical position other in horizontal position.

Giemsa-stained thin blood smear slides from 150 P. falciparum-infected and 50 healthy patients were gathered and photographed at Chittagong Medical College Hospital, Bangladesh. The images annotations were done manually by an expert slide reader at the Mahidol-Oxford Tropical Medicine Research Unit in Bangkok, Thailand. The de-identified images and annotations are archived at NLM. Ersoy et al. [27] used a level-set based algorithm for detecting and segmenting the RBCs.

Some patients may have more than one image associated with their file; in this case each image was treated as a separate patient for purposes of training and prediction. The dataset was collected from U.S National Institute of Health (NIH).

The immediate observation from the images in fig.5., is that parasitized images cells seems to have these different colored spots or patches, while the uninfected ones seems to be uniform and clear.

As shown by the fig.6., the left figure shows that the ratio of parasitized and uninfected data is one in the same file in equal way 50-50. For the right-side fig.6., RBCs malaria dataset was split into two parts, 75% to be used for training and 25% to be used for testing; where ELM model was used together with other ML algorithms like SVM, KNN, CART, RF, CNN, VGG16, RESNET, and DENSENET for performance evaluation of the results.

C. Results and Discussion

1) Representation of the obtained accuracy results for each model

As shown by the table II. of results, ELM outperformed all other algorithms with an accuracy of 99.0% and a shorten time of $28 seconds$.

TABLE II
TABLE OF MODELS ACCURACY AND TIME TAKEN

Name of Classifier	*Accuracy %*	*Time used/seconds*
SVM	68.2	479
KNN	78.3	44
CART	92.3	30
RF	95.5	31
CNN	95.9	1200
VGG16	96.6	1356
RESNET	96.9	1328
DENSENET	97.1	1040
ELM	99.0	28

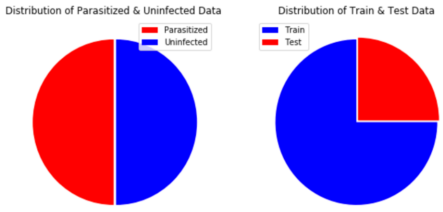

Fig. 6. Splitting images into training and testing.

For these algorithms, the same preprocessing method and features extraction applied are the same. Fig.7. shows the comparison of the accuracy of different used models.

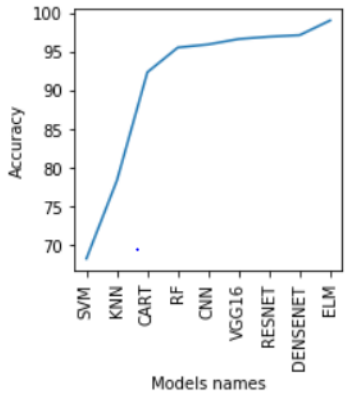

Fig. 7. Representation of accuracy of used models.

2) Representation of obtained confusion matrixes

These confusion matrixes are describing the prediction performance of the applied models for comparison which are DenseNet and ELM as the ones with the highest accuracy on used data. In this paper, 2 labeled classes were predicted which are infected (parasitized) and uninfected RBCs malaria images. Fig.8. shows the confusion matrix of DenseNet models.

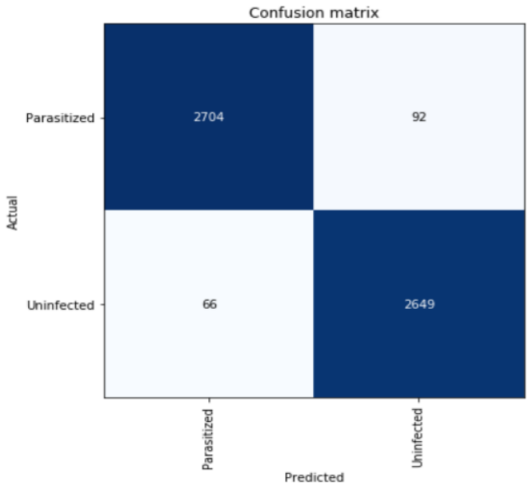

Fig. 8. Representation of confusion matrix results based on DenseNet Model.

From Fig.8., the DenseNet classifier predicted a total of $5511 = (2704+92+66+2649)$ predictions where the diagonal elements are the true predictions and off diagonal elements shows where the classifier went wrong. The DenseNet predicted that among 2796 predictions, 2704 RBCs images are infected by malaria and 92 are not even if instead 92 are infected too. For Uninfected part, the DenseNet predicted that among 2715 predictions, 66 RBCs images are infected instead are uninfected and come up with 2649 RBCs images which its predicted well that are uninfected.

Fig.9. shows the confusion matrix of ELM models.

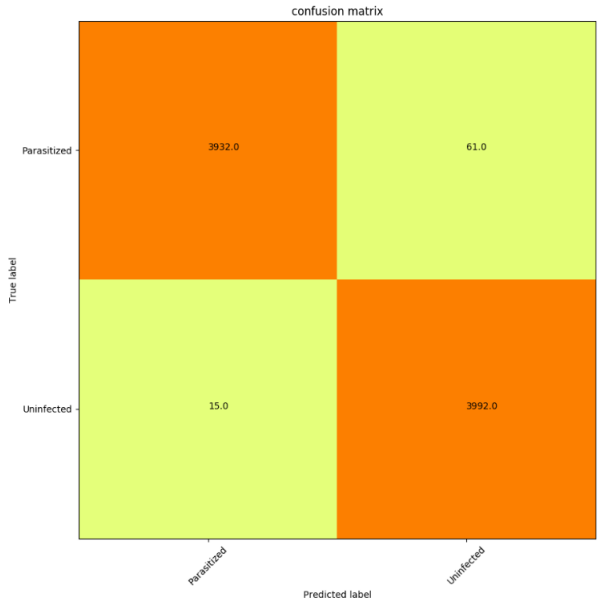

Fig. 9. Representation of confusion matrix results based on ELM Model.

This confusion matrix is describing the prediction performance of the applied model which is ELM on used data which are known. In this paper, 2 labeled classes were predicted which are infected (parasitized) and uninfected RBCs malaria images. Through the fig.9. of confusion matrix obtained using ELM, the ELM classifier predicted a total of $8000 = (3932 + 61 + 15 + 3992)$ predictions where the diagonal elements are the true predictions and off diagonal elements shows where the classifier went wrong. For parasitized/infected part, the ELM predicted that among 3993 predictions, 3932 RBCs images are infected by malaria and 61 are not even if instead 61 are infected too. For Uninfected part, the ELM predicted that among 4007 predictions, 15 RBCs images are infected instead are uninfected and come up with 3992 RBCs images which its predicted well that are uninfected.

Using the two confusion matrixes, different performance evaluation measures were calculated and compared as shown in TableIII.

TABLE III
TABLE OF MODELS ACCURACY AND TIME TAKEN

No.	*Measure Name*	*DensNet*	*ELM*
1.	Accuracy	0.9713	0.9905
2.	Misclassification	0.0287	0.0095
3.	Precision	0.97	0.98
4.	Prevalence	0.5	0.5
5.	Null Error Rate	0.49	0.51
6.	Recall	0.97	098
7.	Specificity	0.97	0.99
8.	False Positive Rate	0.034	0.015

According to the above results in table III. ELM shows that it performs extremely where misclassification error and false positive rate are actual insignificant; precision, specificity, and Recall are high which show how ELM outperform better than other approaches.

3) Prediction

Fig.10. shows the prediction of parasitized and uninfected RBCs images together with the prediction probability. Example the first image show that it is uninfected with a probability of 7.62 and a probability of 0.00 of being parasitized. The second one from left to right side on the first line shows that it is parasitized with a probability of 6.91 and a probability of 0.00 of being uninfected.

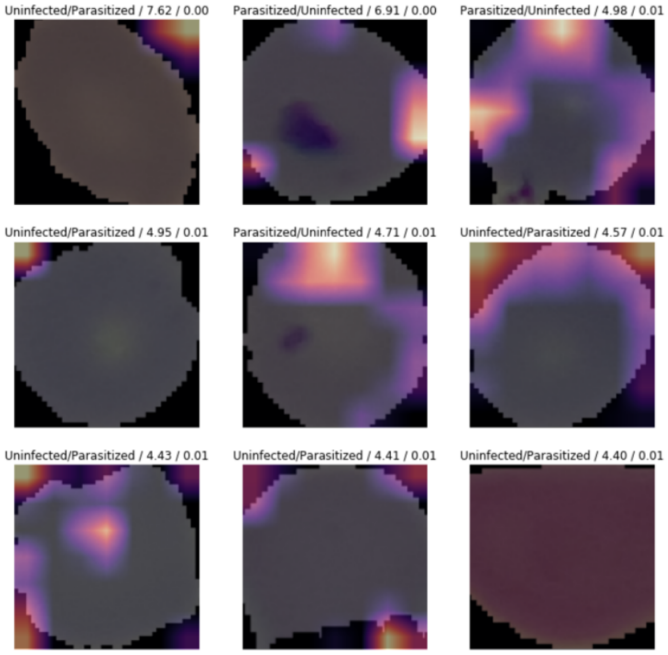

Fig. 10. the prediction of parasitized or uninfected image together with probability.

IV. Conclusion

As mentioned above, this paper has successfully fulfilled its goals with an important contribution over existing machine learning technology where ELM outperforms all existing models on malaria RBC disease prediction with an accuracy of 99.0 and within a short time used of 28 seconds. ELM algorithm is one of the most efficient ML algorithms in the neural networks' world and it works on very large datasets. Because of the non-iterative training all the parameters are tuned once, and this results in a high training speed. Its implementation is easy to understand, and it can be used to solve complex problems in related area. Within the advantages of ELM and its exceedingly performance compared with other ML, its proof that ELM is highly recommend being used in Malaria RBCs classification and prediction.

This paper focused on improving existing models and results, which leads to recommend the next research to differentiate plasmodium of malaria RBCs with other Hyperparasitemia located in RBCs in terms of malaria disease level and trace as well the malaria growth by considering RBCs image per patient not in general, for the future work

V. Acknowledgment

This work is supported by Beijing Institute of Technology.

References

[1] N. Abbas, T. Saba, D. Mohamad, A. Rehman, A. Almazyad, and J. Saleh Al-Ghamdi, "Machine aided malaria parasitemia detection in giemsa-stained thin blood smears," *Neural Computing and Applications*, vol. 29, no. 3, p. 803–818, 2016.

[2] A. Capone, I. Ricci, C. Damiani, M. Mosca, P. Rossi, P. Scuppa, E. Crotti, S. Epis, M. Angeletti, M. Valzano, L. Sacchi, C. Bandi, D. Daffonchio, M. Mandrioli, and G. Favia, "Interactions between asaia, plasmodium and anopheles: New insights into mosquito symbiosis and implications in malaria symbiotic control," *Parasites & vectors*, vol. 6, p. 182, 2013.

[3] I. Bates, V. Bekoe, and A. Asamoa-Adu, "Improving the accuracy of malaria-related laboratory tests in ghana," *Malaria journal*, 12 2004.

[4] F. T. Boray, G. D. Andrew, and K. Izzet, "Computer vision for microscopy diagnosis of malaria," *Malaria Journal*, pp. 1–14, 2009.

[5] S. Akanksha, S. Sini, and D. Shatendra, "Recent image enhancement techniques: A review," *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 4, no. 1, p. 40–45, 2014.

[6] S. Jalari, E. B, K. Reddy, and C.-H. Lai, "An image processing approach for accurate determination of parasitemia in peripheral blood smear images," *International Journal pf Computer Applications*, vol. 1, pp. 23–28, 2011.

[7] O. Nobuyuki, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[8] S. N. Mark and S. A. Alberto, "Feature extraction and image processing," *The ACM Digital Library*, 2008.

[9] L. Zhaohui, P. Andrew, E. IIker, P. Mahdieh, S. Kamolrat, P. Kannappan, G. Peng, H. Md Amir, M. Richard, James, H. Jimmy, Xiangji, J. Stefan, and T. George, "Cnn-based image analysis for malaria diagnosis," in *2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, 2016.

[10] D. Yuhang, J. Zhuocheng, S. Hongda, and P. W. David, "Classification accuracies of malaria infected cells using deep convolutional neural networks based on decompressed images," in *SoutheastCon 2017*. IEEE, 2017.

[11] H. Jane, R. Deepali, C. L. Stefanie, R. Gabriel, N. Francois, A. N. Odailton, M. Benoit, V. L. Marcus, U. F. Marcelo, R. Laurent, T. D. Manoj, T. C. Fabio, M. Matthias, and E. C. Anne, "Applying faster r-cnn for object detection on malaria images," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2017.

[12] M. Leila, A.-A. Karim, and B. Abdolamir, "Malaria parasite detection in giemsa-stained blood cell images," in *2013 8th Iranian Conference on Machine Vision and Image Processing (MVIP)*. IEEE, 2013.

[13] W. Sri and Wijiyanto, "Texture analysis to detect malaria tropica in blood smears image using support vector machine," *International Journal of Innovative Research in Advanced Engineering (IJIRAE)*, vol. 1, no. 8, pp. 302–306, 2014.

[14] G. Lucy, M. M. Daniel, A. K. Kenneth, A. C. K. Mjomba, and S. M. Njogu, "Determination of plasmodium parasite life stages and species in images of thin blood smears using artificial neural network," *Open Journal of Clinical Diagnostic*, vol. 4, pp. 78–88, 2014.

[15] D. K. Das, M. Ghosh, M. Pal, A. Maiti, and C. Chakraborty, "Machine learning approach for automated screening of malaria parasite using light microscopic images," *Micron(Oxford, England : 1993)*, vol. 45, pp. 97–106, 2012.

[16] S. Pradeepa Premaratne, D. Nadira, N. Karunaweera, S. Fernando, W. Supun, R. Perera, R. P Asanga, and S. Rajapaksha, "A neural network architecture for automated recognition of intracellular malaria parasites in stained blood films," 2019.

[17] Z. Huang and J. Leng, "Analysis of hu's moment invariants on image scaling and rotation," in *2010 2nd International Conference on Computer Engineering and Technology*, vol. 7. IEEE, 2010, pp. V7–476.

[18] R. M. Haralick, K. Shanmugam *et al.*, "Textural features for image classification," *IEEE Transactions on systems, man, and cybernetics*, no. 6, pp. 610–621, 1973.
[19] G.-B. Huang, D. Hui Wang, and Y. Lan, "Extreme learning machines: a survey," *International Journal of Machine Learning and Cybernetics*, vol. 2, pp. 107–122, 2011.
[20] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew *et al.*, "Extreme learning machine: a new learning scheme of feedforward neural networks," *Neural networks*, vol. 2, pp. 985–990, 2004.
[21] G.-B. Huang, "What are extreme learning machines? filling the gap between frank rosenblatt's dream and john von neumann's puzzle," *Cognitive Computation*, vol. 7, pp. 263–278, 2015.
[22] H. Guang-bin, Z. Qin-yu, and S. Chee-kheong, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, no. 1-3, pp. 489–501, 2006.
[23] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics : a publication of the IEEE Systems, Man, and Cybernetics Society*, vol. 42, pp. 513–529, 2011.
[24] G. Huang, G.-B. Huang, S. Song, and K. You, "Trends in extreme learning machines: A review," *Neural Networks*, vol. 61, pp. 32–48, 2015.
[25] P. Pandey and R. G. Singh, "Analysis of randomized performance of bias parameters and activation function of extreme learning machine," *International Journal of Computer Applications*, vol. 135, no. 1, pp. 23–28, 2016.
[26] A. Akusok, K.-M. Björk, Y. Miche, and A. Lendasse, "High-performance extreme learning machines: a complete toolbox for big data applications," *IEEE Access*, vol. 3, pp. 1011–1025, 2015.
[27] I. Ersoy, F. Bunyak, J. Higgins, and K. Palaniappan, "Coupled edge profile geodesic active contours for red blood cell flow analysis," in *In Proceedings of the 9th IEEE international symposium biomedical imaging*. IEEE, 2012, p. 748–751.