Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 3922763, 11 pages
https://doi.org/10.1155/2022/3922763

Research Article

Detection of Peripheral Malarial Parasites in Blood Smears Using Deep Learning Models

Amal H. Alharbi,[1] Aravinda C. V ,[2] Meng Lin ,[3] B Ashwini,[2] Mohamed Yaseen Jabarulla ,[4] and Mohd Asif Shah [5]

[1]*Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia*
[2]*N. M. A. M. Institute of Technology, Nitte 574110, Karkala, India*
[3]*Ritsumeikan University, Kyoto, Japan*
[4]*School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology (GIST), Gwangju, Republic of Korea*
[5]*Kebri Dehar University, Kebri Dehar, Ethiopia*

Correspondence should be addressed to Mohd Asif Shah; ohaasif@kdu.edu.et

Received 1 April 2022; Revised 17 April 2022; Accepted 25 April 2022; Published 24 May 2022

Academic Editor: Arpit Bhardwaj

Due to the plasmodium parasite, malaria is transmitted mostly through red blood cells. Manually counting blood cells is extremely time consuming and tedious. In a recommendation for the advanced technology stage and analysis of malarial disease, the performance of the XG-Boost, SVM, and neural networks is compared. In comparison to machine learning models, convolutional neural networks provide reliable results when analyzing and recognizing the same datasets. To reduce discrepancies and improve robustness and generalization, we developed a model that analyzes blood samples to determine whether the cells are parasitized or not. Experiments were conducted on 13,750 parasitized and 13,750 parasitic samples. Support vector machines achieved 94% accuracy, XG-Boost models achieved 90% accuracy, and neural networks achieved 80% accuracy. Among these three models, the support vector machine was the most accurate at distinguishing parasitized cells from uninfected ones. An accuracy rate of 97% was achieved by the convolution neural network in recognizing the samples. The deep learning model is useful for decision making because of its better accuracy.

1. Introduction

As per the World Health Organization, 3.4 million inhabitants in 92 countries may be at risk of malaria infection, with 1.1 billion people at high risk. Epidemiological factors can affect the transmission of malaria, including an ecological and epidemiological study performed in a confined, isolated location [1]. As a result, the WHO supports the advancement of rapid and inexpensive diagnostic testing that aids incorrect treatment method identification. In 2019, there has been an estimated 229 million cases of malaria worldwide [2, 3]. The rate of death totaled 409,000 cases of malaria, as measured by an annual survey [4]. Malaria is caused by parasites transmitted through mosquito bites from infected female Anopheles mosquitoes [5, 6]. In general, microscopy testing is widely accepted and widely used for attracting potential patients with malaria. Children under five are most likely to be affected by malaria [7, 8]. The WHO report predicts that the African region carries the highest share of the global malaria burden.

As per WHO research, the African region might feel the consequences of the global malaria burden [9]. Since it would provide more accuracy, improve consistency, and be cost effective in rural regions, an automated malaria diagnosis system would make a big difference in eliminating the shortage of this insufficiency [10, 11]. Thus, according to medical specialists from the World Health Organization, several malaria parasite groups may cause malaria infection in humans [12]. These include "Plasmodium Falciparum, Plasmodium Vivax, Plasmodium Malaria, Plasmodium Ovalle, and Plasmodium-knowlesi." Out of these, the most common two classes are " Plasmodium Falciparum and Plasmodium Viva."

Figure 1 illustrates how malarial cells progress through their various stages: an examination of the first slide shows trophozoites and gametocytes of P. falciparum along with white blood cells as mentioned in Figure 2. A comparison is now made between the enlarged nucleus and the rest of the red blood cells. The second image proves how Plasmodium Falciparum is erected with Plasmodium Schizonts. There is a need for researchers to investigate innovations relating to malaria diagnosis to automate the process for society. This field has seen an increase in good research articles over several decades. Besides research articles, automatic malaria diagnosis has occurred with a variety of software tools and hardware tools.

2. Related Work

Makhija et al. implemented a V-value histogram method. It has achieved 60% sensitivity [1]. Rajaraman et al. designed the contrast enhancement and threshold-based segmentation approach. A qualitative analysis of two hundred patients' images was achieved using this method [2]. Suriya et al. worked on the color information-based pattern segmentation by considering 75 patients' image samples and achieved around 90% of detection [3]. Liang et al., according to their study, could detect 90% of the speckle noise images using a median filter based on histogram and morphological operations [4]. Vijayalakshmi et al. worked on microscopic images by applying transfer learning on VGG 16 and SVM classifier and claiming the result of 91% of classification accuracy [5].

Hommelsheim et al. implemented the DNA/RNA-binding domains with nucleotide sequence accuracy and redesigned transcription-activator-like signaling pathways to alter genomes with improved binding specificity [6]. Hawkes et al. worked on an analysis of rapid diagnostic tests and made the awareness of utilizing the RDT for equality and ease of use for cost effectiveness [7]. Ross et al. implemented a method for automating the diagnosis of malaria from thin blood smears that has been developed. Identification of infected erythrocytes is achieved with 75% sensitivity and an 85% positive predictive value (PPV) [8]. Das et al. worked on parasite characterization and classification using ML on light microscopic images of peripheral blood smears and achieved 89% classification using an SVM classifier [9].

Postiche et al. discussed the development of image analysis and machine learning for the diagnosis of malaria microscopically as well as the emergence of smartphone technology for future diagnosis [10]. LeCun et al. worked to find complex structures in large datasets, and the backpropagation algorithm was implemented to propose that a machine could constantly update its internal parameters based on its representation in the prior phases. [11]. Dahou Yang et al. compared the detection of antimalarial efficacy of an image-based cytometer with a commercial flow cytometer and the results of these two tests [12].

Yunda et al. determined that by applying principal component analysis to samples of thick film blood, they could reduce the number of features. [15]. Kaewkamnerd et al. to detect the sensitivity of plasmodia on thick blood films, created a two-stage algorithm [16]. Hanif et al. developed an elongating technique for the enhancement and segmentation of heavy blood smear images of Plasmodium falciparum. They reduced the amount of noise and blurring while increasing the contrast range to make the images more visible [17]. The existing models suffer from various issues such as overfitting [13, 18, 19], vanishing gradient [20–22], and poor convergence speed [23–25] kinds of problems.

3. Overview of Convolution Neural Network (CNN)

Convolution layers comprise the CNN (convolutional neural network). Pixel-by-pixel weights and biases are learned. The synapse cell accepts a variety of inputs before combining the weights, either from the input in each layer or from the weights and biases passing through the activation node. This level comprises the input pixels and weights that are shared by the layer's neurons [26–28]. Furthermore, the inputs to standard neural networks are vectors of a single dimension, but in CNN, they are represented as multichannel images. It uses the framework of this algorithm to optimize such algorithms as random gradient, Adam, AdaGrad, etc. as well as to perform tasks such as object detection, image classification, and localization as mentioned in Figure 3.

3.1. Pooling Layers. Filters are applied to the input images to generate feature maps that distinguish the inclusion of those features and they accomplish it by pooling layers. The constraint of this feature map is that it will show a feature's location precisely. The feature map has the dimensions $nh * nw * nc$, and the output obtained after the pooling layer is $(nh - f + 1)/s * (nw - f + 1)/x * nc$. There are three types of pooling layers such has, Max, Averaging, and Global Pooling. Feature pooling takes the maximum value from a region on the feature map. Average pooling takes the average of the elements present in the neighborhood. The Global pooling takes the value of the entire feature map and scales it down.

3.2. Frequently Used Activation Layer

Sigmoid: the mathematics of the sigmoid function involves the outcome of taking a single number and achieving convinced mathematical operations. $\sigma\ (x) = 1/(1 + e^{-x})$. This takes the real input number and flattens the range between 0 and 1.

Tanh: in the study, tanh is a nonlinearity that is preferred over the sigmoid functions. Even though it

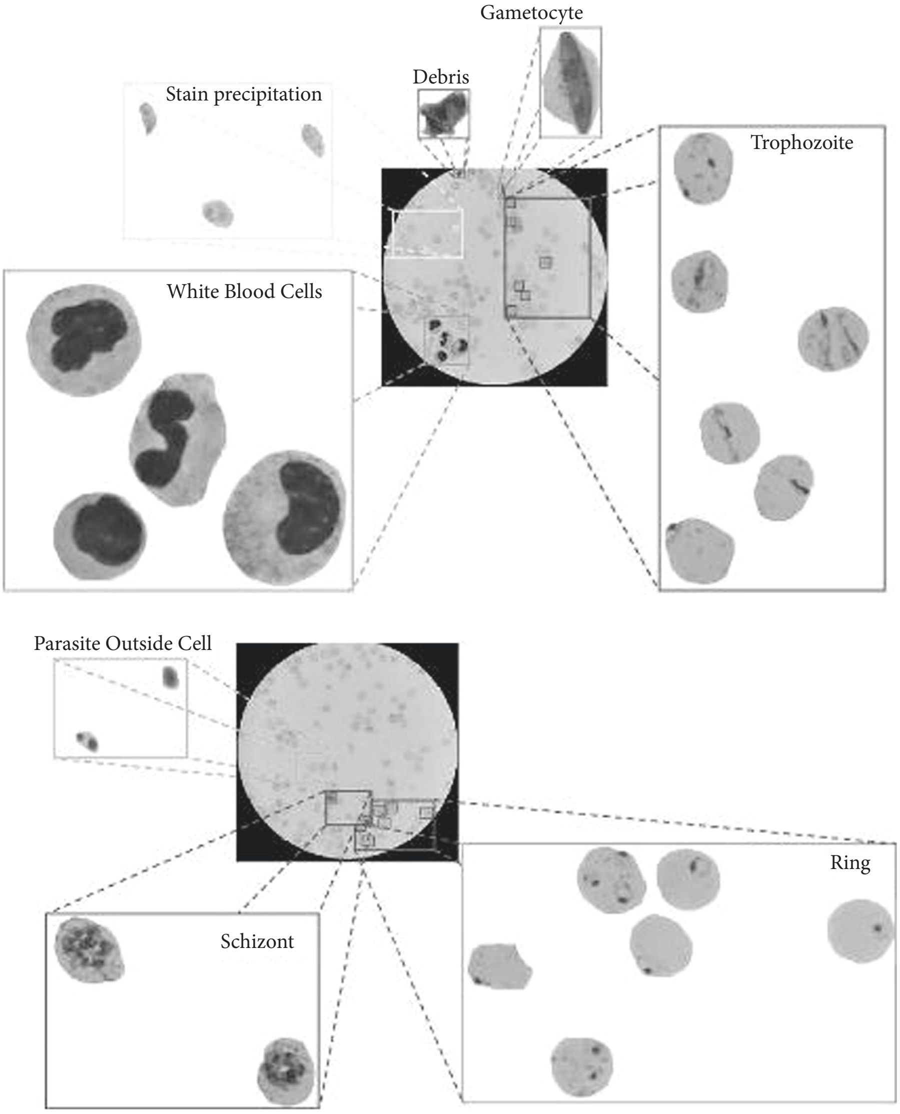


Figure 1: Stages of malaria in blood smear [13].

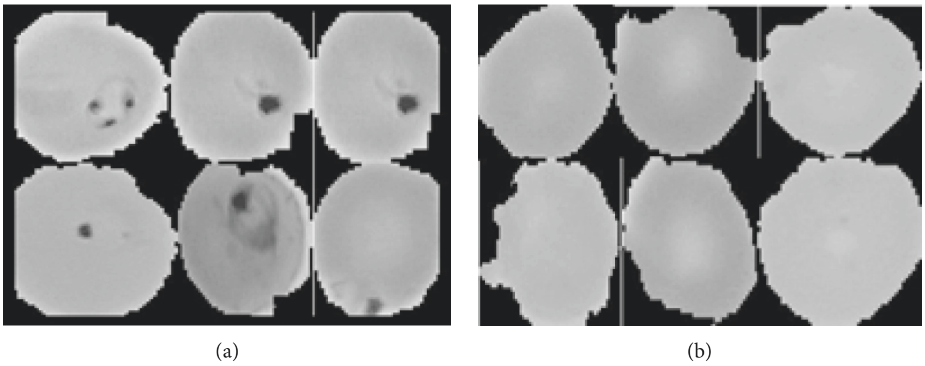
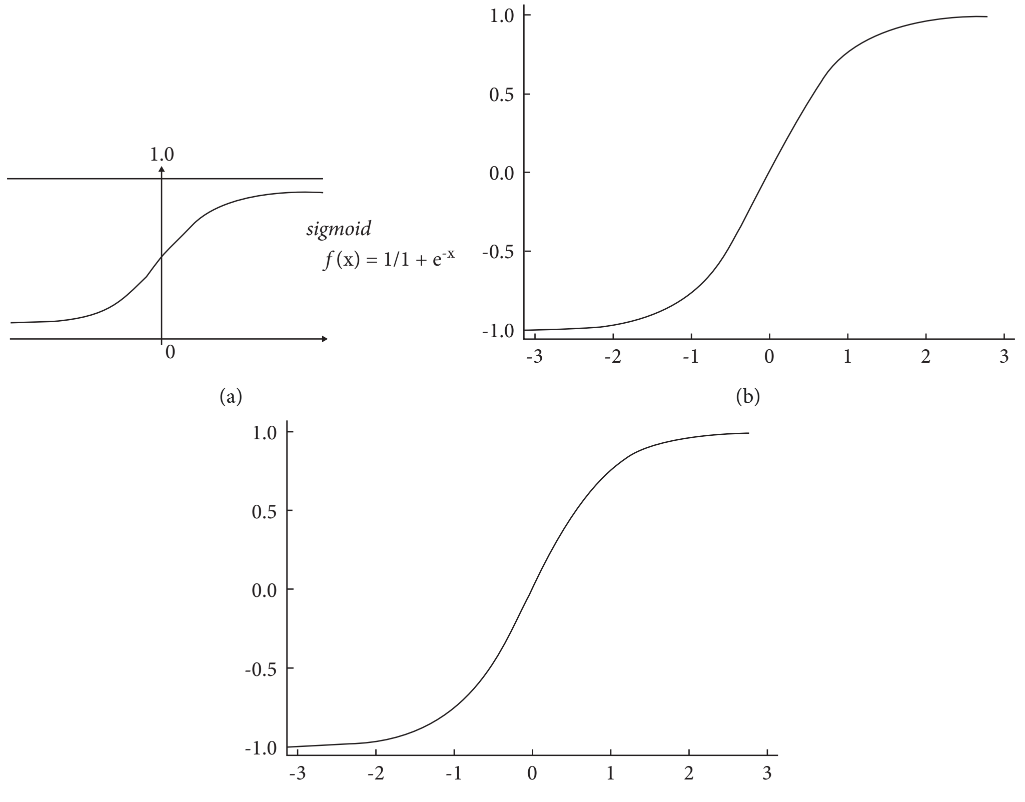


Figure 2: Affected samples and clean samples [14].



(a)

(b)

(c)

Figure 3: Example of a two-part figure with individual subcaptions showing that the captions are sigmoid nonlinearity range between [0, 1] and tanh nonlinearity squashes real numbers to the range between [−1, 1]and ReLU is which is zero when $x < 0$ and then the linear with slope 1 when $x > 0$. (a) Sigmoid, (b) Tanh, and (c) ReLU.

shows better results than the sigmoid functions, it is still not possible to solve gradient problems using the Tanh function. $\tanh(x) = 2\sigma(2x) - 1$.

ReLU: as said in the function, the activation is simply a threshold at zero. Tanh functions are mathematically portrayed as $f(x) = \max(0, x)$.

4. Methodology

Various examination of the freely accessible datasets was utilized for classification, augmentation, and preprocessing. As shown in Table 1, a few authors have already provided a dataset that we evaluated for collection, classification, augmentation, and preprocessing techniques.

4.1. Datasets. Figure 2 illustrates two sets of datasets consisting of roughly 13,000 samples, one parasitized and one nonparasitic, illustrating infected and uninfected malarial blood samples.

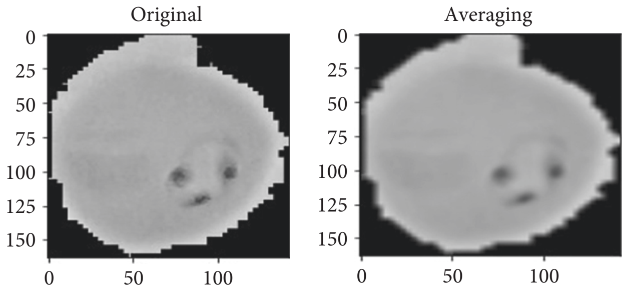
4.2. Classification of Malaria Cells. The use of computer vision and machine learning algorithms to diagnose malaria has recently got performance metrics in plenty of new studies. As part of a collaborative effort, a recently proposed automated system for detecting and acting on red blood cells was recently presented.

Table 1: Existing study of methodologies.

Model	Methodology
Rajaraman et al. [2]	CNN with 2-level segmentation
Narayanan et al. [29]	Encoder-decoder architecture
Tran et al. [9]	Segmentation using deep learning
Li et al. [30]	Deep CNNs for HEp-2 cell classification
Das et al. [31]	Bayesian learning and support vector machine

4.3. Image Smoothing. The impact of blurring was applied to several smoothing algorithms, such as Gaussian noise, salt pepper noise, and bilateral filters for both noisy images, and this was compared using the Gaussian, median, and bilateral filters. By eliminating noise and blurring an image, 2D convolution filtering utilized in various low-pass and high-pass filters achieved promising results. The high-pass filter recognized the edges in a cell image and generated promising results. For this cell image, a 2 averaging filter kernel has been used, $K = 1/9$ as shown in Figures 4 and 5.

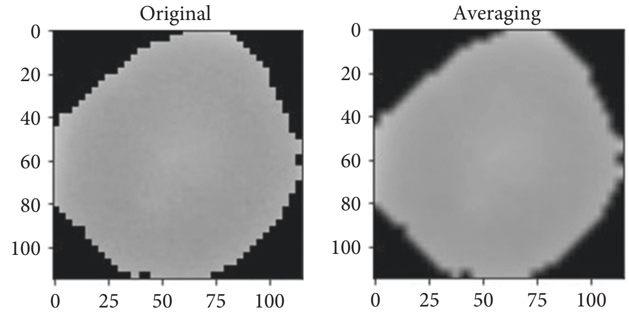
4.4. Gabor Filtration Technique. The presence of no malaria-infected cells in many samples allows them to be used as a method to reduce the overall processing run time, which is why statistical analysis is applied to calculate the infected cell



(a)

(b)

Figure 4: Two parts with separate captions for smoothing the original infected sample and smoothing averaging the original infected sample. (a) Infected original sample. (b) Image greyscale.



(a)

(b)

Figure 5: Two parts with separate captions for smoothing the uninfected original sample and smoothing averaging the original sample. (a) Infected original sample. (b) Infected original sample.

sample numbers. A threshold was applied using the Gabor filter method to color the image of the infected area after this was noticed. Distortion was detected in both the background and inside RBCs, according to results of this approach. Later, morphological series have been used to fill the gaps to gain distinct samples, as illustrated in Figure 6. The precision of a Gabor filter influences its orientation; this has kernels that are common to the 2D field and depict essential spatial localization and orientation aspects; as a result, the kernels of this filter are also relative term of dimensions mentioned in (1).

$$\psi_{\omega\phi} = \frac{\omega}{\sqrt{2\pi C}} \frac{e^{-} \omega^{2}\left(4ap^{2} + bq^{2}\right)}{\left(8c^{2}\right)\left(e^{i} a\omega - e^{-} c^{2}/2\right)}. \tag{1}$$

The actual and imagined regions of the filtration are shown in Figure 4. (infected) and Figure 5. (uninfected). We assume that the value of I(ap + bq) is a grey value at (ap, bq). The sample convolution and the scale's Gabor kernel, as well as the direction of θ, are specified as

$$G_{\omega,\phi} = I \otimes \psi_{\omega}\phi. \tag{2}$$

Equation (3) now has two actual and imagined values. Each separated by a distance orientation's response is specified as

$$I_{\omega}\theta(z) = R_{e}(G, \omega, \theta(z))^{2} + \mathrm{Im}(G, \omega, \theta(z))^{2}. \tag{3}$$

Figures 6 and 7 illustrates images combined with subsampled values analyzed with Gabor. The values considered k-size = 20*20, $\sigma = 4$, $\theta = 1^{*} np.pi/2$, lamda (λ) = $1^{*} np.\ pi/4$, $\phi = 0.8$.

4.5. Data Preprocessing. The model's actions depend on the data that were provided in the supervised process of learning. As a result, it has a profound influence on decision making. A large dataset is obtained because the image smoothing algorithm is applied for feature extraction. The vector was standardized in the range of 0 to 255 as the first step toward accurate data identification. The chi-square feature selection method was used for selecting 80% of the most useful features for the final vectors.

5. Classifications

5.1. Support Vector Machine. A support vector machine (SVM) is useful for deciding the optimal location of a decision boundary or for learning statistics and for determining when it is necessary to separate classes. Regarding multiclass classification problems, this study used the "each" strategy, wherein labels are assigned from a finite set of several elements. Infectious parasite samples and unaffected parasitic samples are two classes that were used to estimate

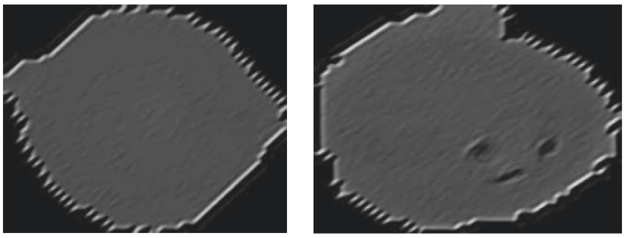
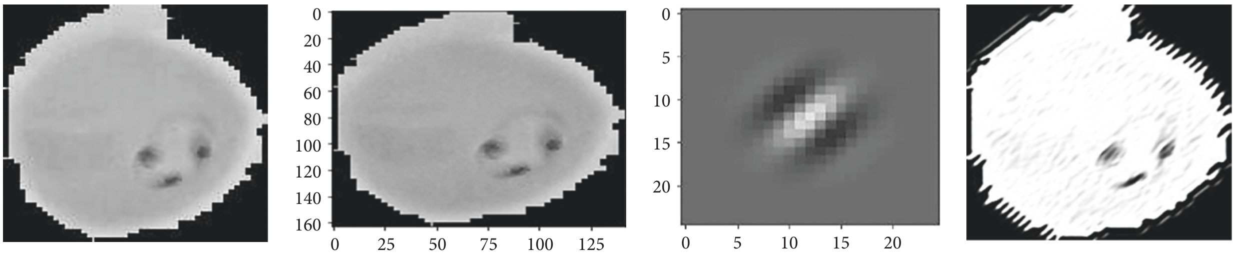

(a) (b)

Figure 6: Infected- Gabor-filtered and the other uninfected Gabor-filtered. (a) Affected initial sample. (b) A greyscale sample.



(a) (b) (c) (d)

Figure 7: Infected kernel sample, infected greyscale, and an output image that is resized according to its kernel size. (a) Infected sample, (b) Gray scale, (c) Gabor filter image, (d) Output generated.

the number of classes. The "one-to-one" classifier results can be transformed to form a decision function of shape by using the decision function shape option, $(13000\text{samples}, 2\text{classes})$. The test vectors are analyzed by applying each classifier to them, giving each vote.

5.2. XG-Boost Algorithm. Gradient boosting reduces the loss function by building models from single weak learners in an iterative fashion instead of building a complete model from random subsets or features like the random forest. The loss function for gradient boosting is minimized using gradient descent. Based on the application of this algorithm, it is possible to achieve a score of about 90% in accuracy and speed because it uses advanced regularization techniques to prevent overfitting and speed up computation, as shown in Table 2.

Table 2: 94% accuracy in support vector machine classification.

Precision measure	Recall measure	$F1$-score (%)
Parasitized-results-0.9259	0.9615	0.9433
Infected-results-0.9565	0.9166	0.9361

5.3. Proposed Convolutional Neural Network. A trainable customizable network, which is divided into four fully connected layers, is based on the convolution layer, the pooling layer, and the average pooling step to obtain features, reduce computations, and update features. 26,188 samples were divided into three sets: training, testing, and validation. It was then converted to a 128×128 pixel size. According to Figures 8 and 9, the study included 13,779 uninfected samples.

6. Evaluation Metrics

6.1. Optimizer Function. In contrast to the RMSProp optimizer function, the Adam optimizer could much more successfully overcome the AdaGrad optimizer's inadequacy. It is a much more advanced version of stochastic gradient descent wherein the weights are continuously updated with the training data. The iterations must be stated in Step 1 for the algorithm to work. The very first step is to get a gradient. Step 2 is by using the moving averages formula for determining the moving averages. In this step, the estimator's bias is corrected, where *xhat* and *yhat* are biased correction equations, and in the last step of the Adam algorithm, the weights are adjusted in the network using the z-expression.

6.2. Loss Function. In the optimizer, the neural net's weights and biases are updated to reduce the loss function. Loss functions are defined as ways of mapping input functions to output functions. This function will decide the probability value of a prediction class by calculating the categorical cross-entropy, as shown in (4).

$$L(\widehat{y, y}) = -\frac{1}{N}\frac{N}{in}\frac{ca}{ca=1}1yi_{\varepsilon}C_{ca}\log P_{m}[y_{i}\varepsilon c_{ca}], \qquad (4)$$

where "ca" is the number of categories and "in" is the number of observations.

6.3. Performance Analyses

6.3.1. Experimental Set-Up and Performance Metrics. On the server, the recognition system is installed for online access. The CPU is a Lenovo ThinkPad workstation, RAM is 32G, and the operating system is Windows 10.

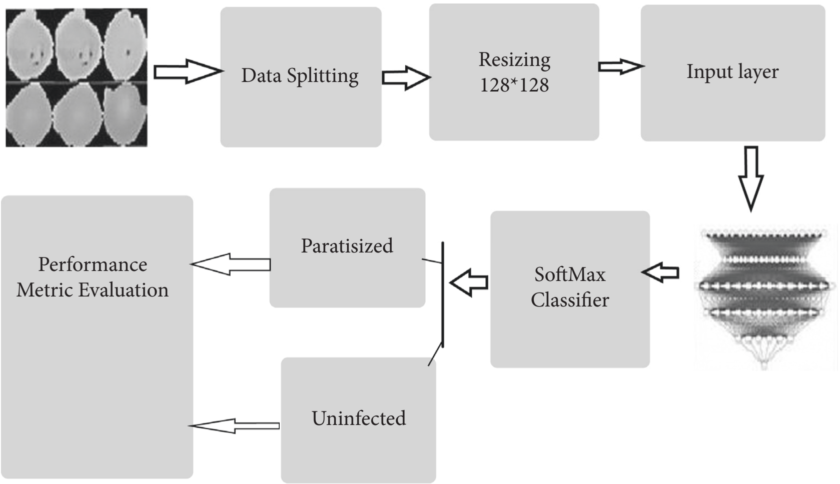
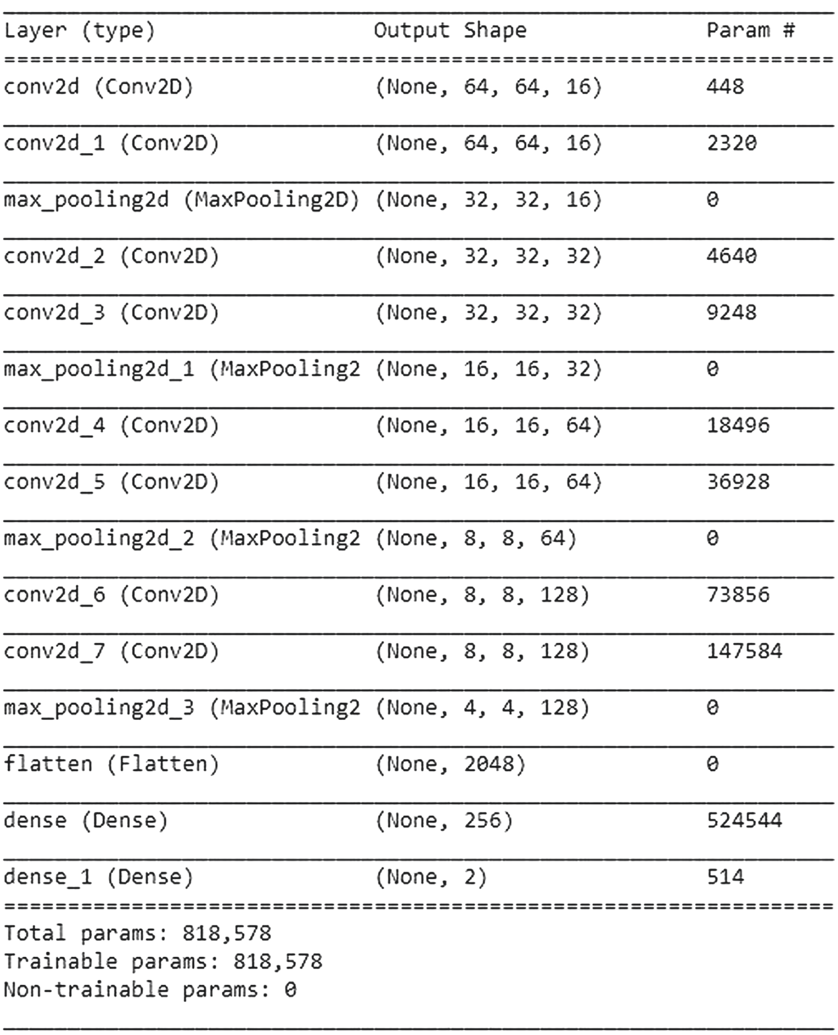


Figure 8: Proposed block diagram of CNN.



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 64, 64, 16)        448
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 64, 64, 16)        2320
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 32, 32, 16)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 32, 32, 32)        4640
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 32, 32, 32)        9248
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 16, 16, 32)        0
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 16, 16, 64)        18496
_________________________________________________________________
conv2d_5 (Conv2D)            (None, 16, 16, 64)        36928
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 8, 8, 64)          0
_________________________________________________________________
conv2d_6 (Conv2D)            (None, 8, 8, 128)         73856
_________________________________________________________________
conv2d_7 (Conv2D)            (None, 8, 8, 128)         147584
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 4, 4, 128)         0
_________________________________________________________________
flatten (Flatten)            (None, 2048)              0
_________________________________________________________________
dense (Dense)                (None, 256)               524544
_________________________________________________________________
dense_1 (Dense)              (None, 2)                 514
=================================================================
Total params: 818,578
Trainable params: 818,578
Non-trainable params: 0
_________________________________________________________________
```

Figure 9: Architecture configuration set-up.

```
for i in range (num_iterations):
p = compute-the-gradient(m, n)
xx = $\beta 1*xx + (1 - \beta 1) * w$
yy = $\beta\_2 * xx + (1 - \beta 2) * \mathrm{power}(y, 2)$
    $xhat = m/(1 - \mathrm{np.power}(\beta 1, i)) + (1 - \beta 1) * w/(1 - \mathrm{np.power}(\beta 1, i))$
    $yhat = y/(1 - \mathrm{np.power}(\beta 2, i))z$
    $= z - size * xhat/(np.\sqrt{yhat} + \varepsilon)$
```

Algorithm 1: Adam optimizer algorithm.

Table 3: 90% accuracy of extreme gradient boosting algorithm classification.

Precision measure	Recall measure	$F1$-score (%)
Parasitized-results-0.8275	0.9230	0.8727
Infected-results-0.9047	0.7916	0.8444

Table 4: Proposed CNN with state-of-the-art models, 96%.

Authors work	Methodology	Accuracy in %	Sensitivity	Specificity	$F1$-score
Rajaraman et al. [2]	Pretrained-CNN	0.98	0.981	0.992	0.987
Vijayalakshmi et al. [5]	CNN with 2-level	0.977	0.971	0.972	0.959
Liang et al. [4]	CNN (16 layers)	0.973	0.969	0.977	—
Das et al. [9]	Bayesian learning	0.84	0.981	0.689	—
Ross et al. [8]	Image enhancement	0.73	0.85		
Proposed model	Customized CNN	0.98	0.985	0.988	0.987

The problem consists of sorting the images into two classes: parasitized (i) and uninfected (ii). The $F1$-score is calculated by using the harmonic mean of precision and recall. It can be evaluated as mentioned in (5).

$$F1 - score = 2 * \frac{(\text{precision} \times \text{recall})}{(\text{precision} + \text{recall})}. \tag{5}$$

True positive rate is another well-known measure. Compared to actual positives correctly classified as positives, this reflects the number of true positives, and it can be defined as mentioned in (6).

$$\text{True Positive Rate } = \frac{TP}{(TP + FN)}. \tag{6}$$

Accuracy can be defined as the systematic errors that measure the differences between true and predicted values, and it can be defined as mentioned in (7).

$$Accuracy = \frac{(\text{TP} + \text{TN})}{(P/N)}. \tag{7}$$

The positive predicted value represents the subjects who were positive for the presence of the disease during the screening procedure, and it can be defined as mentioned in (8).

$$PositivePredictedValue = \frac{\text{TP}}{(\text{TP} + \text{FP})}. \tag{8}$$

The negative predicted value represents how likely it is for the subjects screened for the disease to have a negative result, and it can be defined as mentioned in (9).

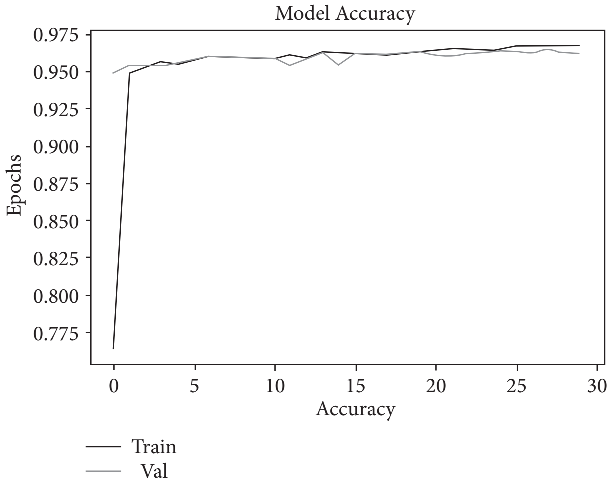


Figure 10: Training and validation accuracy.

$$\text{False Positive Rate } = \frac{FP}{(FP + TN)}. \tag{9}$$

The false negative rate can be calculated by adding the number of positive events that were misclassified as negatives to the total number of positive events, and it can be defined as mentioned in (10).

$$\text{False Negative Rate } = \frac{FN}{(FN + TP)}. \tag{10}$$

To determine the accuracy of malarial parasite detection using ML algorithms, experiments were conducted with several algorithms. Based on Tables 2 and 3, the classification accuracy obtained by the SVM was around 94%, and XG-Boost had an

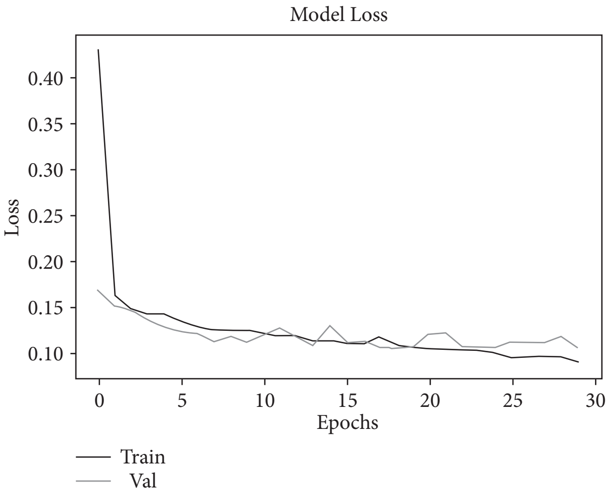


Figure 11: raining and validation loss.

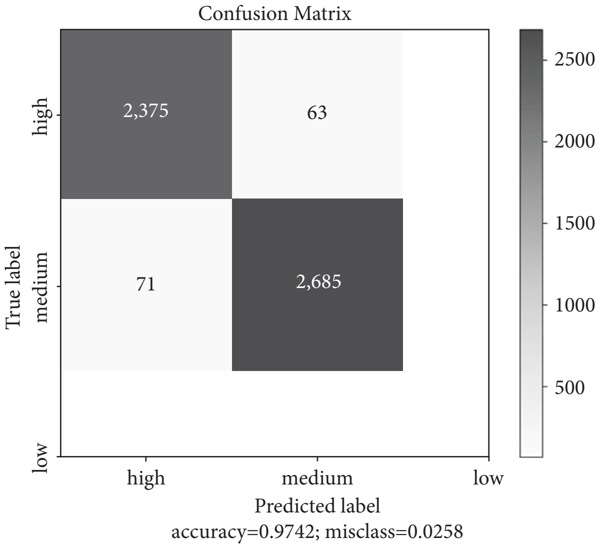


Figure 12: Confusion matrix accuracy 97%.

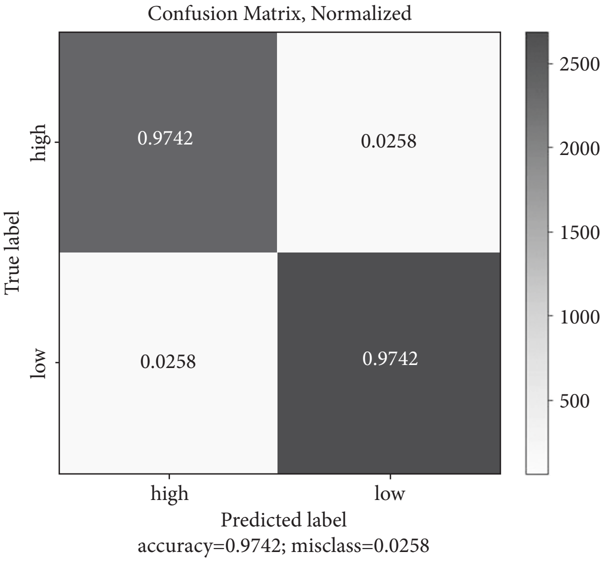


Figure 13: Confusion matrix normalized.

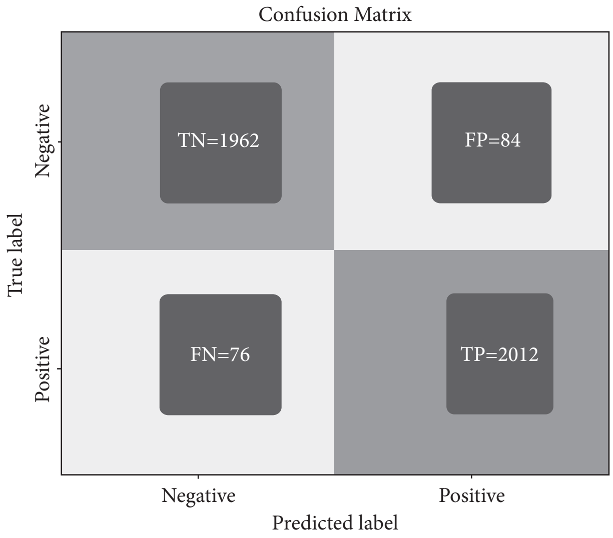


Figure 14: Confusion matrix true positive and true negative.

estimated classification accuracy of 90%. As shown in Table 4, an estimated 98.0% of the data was able to be classified by the convolution neural network. Figures 10 and 11 show the training and validation loss followed by training and validation accuracy, respectively, and Figure 12 the confusion matrix accuracy, Figure 13 confusion matrix normalized, and Figure 14 confusion matrix true positive and true negative, respectively.

7. Conclusion

Plasmodium parasites cause dengue infection in red blood cells. Counting blood samples mechanically is a very time-consuming process that results in a tedious diagnosis strategy. For the detection and analysis of this malarial virus, the XG-Boost classification algorithm, support vector machine, and neural network algorithms were compared using Gabor filters. Convolutional neural networks perform well on the same datasets when analyzing and recognizing them. A model is developed that analyzes the blood sample to determine parasitized or uninfected cells. Using this model, we aim to reduce model discrepancies and improve their robustness and generalization. We collected 13,750 parasitized samples and 13,750 nonparasitic samples for comparative analyses. Support vector machines were accurate to 94%, while XG-Boost achieved 90% accuracy, and neural networks achieved 80% accuracy, respectively. Both parasitized and uninfected cells were more accurately classified by the support vector machine than by the other two models. In either case, convolutional neural networks were designed to recognize the samples with 97% accuracy. In terms of decision making, these models are helpful because of their improved accuracy.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was funded by Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R120), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

References

[1] K. S. Makhija, S. Maloney, and R. Norton, “The utility of serial blood film testing for the diagnosis of malaria,” *Pathology*, vol. 47, no. 1, pp. 68–70, 2015.

[2] S. Rajaraman, S. K. Antani, M. Poostchi et al., “Pre-trained convolutional neural networks as feature extractors toward improved malaria parasite detection in thin blood smear images,” *PeerJ*, vol. 6, Article ID e4568, 2018.

[3] M. Suriya and M. G. Sumithra, “Efficient evolutionary techniques for wireless body area using cognitive radio networks,” *Computational Intelligence and Sustainable Systems*, Springer, Berlin, Germany, pp. 61–70, 2018.

[4] Z. Liang, A. Powell, I. Ersoy et al., “CNN-based Image Analysis for Malaria Diagnosis,” in *Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine. BIBM*, pp. 493–496, Shenzhen, China, December 2017.

[5] A. Vijayalakshmi and B. Rajesh Kanna, “Deep learning approach to detect malaria from microscopic images,” *Multimedia Tools and Applications*, vol. 79, pp. 15297–15317, 2019.

[6] C. M. Hommelsheim, L. Frantzeskakis, M. Huang, and B. Ülker, “PCR amplification of repetitive DNA: a limitation to genome editing technologies and many other applications,” *Scientific Reports*, vol. 4, p. 5052, 2014.

[7] M. Hawkes, J. Katsuva, and C. Masumbuko, “Use and limitations of malaria rapid diagnostic testing by community health workers in the war-torn Democratic Republic of Congo,” *Malaria Journal*, vol. 8, no. 1, p. 308, 2009.

[8] N. E. Ross, C. J. Pritchard, D. M. Rubin, and A. G. Dusé, “Automated image processing method for the diagnosis and classification of malaria on thin blood smears,” *Medical, & Biological Engineering & Computing*, vol. 44, pp. 427–436, 2006.

[9] D. K. Das, M. Ghosh, M. Pal, A. K. Maiti, and C. Chakraborty, “Machine learning approach for automated screening of malaria parasite using light microscopic images,” *Micron*, vol. 45, pp. 97–106, 2013.

[10] 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.

[11] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, pp. 436–444, 2015.

[12] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, pp. 2278–2323, 1998.

[13] D. Singh, V. Kumar, M. Kaur, M. Y. Jabarulla, and H. N. Lee, “Screening of COVID-19 suspected subjects using multi-crossover genetic algorithm based dense convolutional neural network,” *IEEE Access*, vol. 9, pp. 142566–142580, 2021.

[14] A. Rahman, H. Zunair, M. S. Rahman et al., “Improving malaria parasite detection from red blood cell using deep convolutional neural networks,” 2019, https://arxiv.org/abs/1907.10418.

[15] L. Yunda, A. A. Ramirez, and J. Millán, “Automated image analysis method for p.vivax malaria parasite detection in thick film blood images,” *Sist. Telemática*, vol. 10, pp. 9–25, 2012.

[16] S. Kaewkamnerd, A. Intarapanich, M. Pannarat, S. Chaotheing, C. Uthaipibull, and S. Tongsima, “Detection and classification device for malaria parasites in thick blood films,” in *Proceedings of the 6th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems*, pp. 435–438, Prague, Czech Republic, November 2011.

[17] N. S. M. M. Hanif, M. Y. Mashor, and Z. Mohamed, “Image enhancement and segmentation using the dark stretching technique for plasmodium falciparum for thick blood smear,” in *Proceedings of the 2011 IEEE 7th International Colloquium on Signal Processing and its Applications*, pp. 257–260, Malaysia, May 2011.

[18] G. De Luca, “A survey of nisq era hybrid quantum-classical machine learning research,” *Journal of Artificial Intelligence and Technology*, vol. 2, no. 1, pp. 9–15, 2022.

[19] D. Jie, G. Zheng, Y. Zhang, X. Ding, and L. Wang, “Spectral kurtosis based on evolutionary digital filter in the application of rolling element bearing fault diagnosis,” *International Journal of Hydromechatronics*, vol. 4, no. 1, pp. 27–42, 2021.

[20] M. Kaur, D. Singh, V. Kumar, B. B. Gupta, and A. A. Abd El-Latif, “Secure and energy efficient-based E-health care framework for green internet of things,” *IEEE Transactions on Green Communications and Networking*, vol. 5, no. 3, pp. 1223–1231, 2021.

[21] R. Amin and M. Mojarad, “Modeling the scheduling problem in cellular manufacturing systems using genetic algorithm as an efficient meta-heuristic approach,” *Journal of Artificial Intelligence and Technology*, vol. 1, no. 4, pp. 228–234, 2021.

[22] T. V. Hahn and C. K. Mechefske, “Self-supervised learning for tool wear monitoring with a disentangled-variational-autoencoder,” *International Journal of Hydromechatronics*, vol. 4, no. 1, pp. 69–98, 2021.

[23] H. Kaushik, D. Singh, M. Kaur, H. Alshazly, A. Zaguia, and H. Hamam, “Diabetic retinopathy diagnosis from fundus images using stacked generalization of deep models,” *IEEE Access*, vol. 9, pp. 108276–108292, 2021.

[24] J. Liu, Z. Liu, C. Sun, and J. Zhuang, “A data transmission approach based on ant colony optimization and threshold proxy re-encryption in wsns,” *Journal of Artificial Intelligence and Technology*, vol. 2, no. 1, pp. 23–31, 2022.

[25] A. Balakrishna and P. K. Mishra, “Modelling and analysis of static and modal responses of leaf spring used in automobiles,” *International Journal of Hydromechatronics*, vol. 4, no. 4, pp. 350–367, 2021.

[26] Akshaya and C. V. Aravinda, “Predictive analysis of malignant disease in woman using machine learning techniques,” in *Proceedings of the Advances in Artificial Intelligence and Data Engineering*, pp. 431–438, Springer, Singapore, August 2020.

[27] C. V. Aravinda, L. Meng, M. Atsumi, K. R. Udaya, and G. Amar Prabhu, “A complete methodology for kuzushiji historical character recognition using multiple features approach and deep learning Model,” *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 1, no. 8, 2020.

[28] C. V. Aravinda, Z. Meng, S. Kenshi, Y. Duan, U. Kazuaki, and P. G. Amar, “Apathy classification based on doppler radar image for the elderly person,” *Frontiers in Bioengineering and Biotechnology*, vol. 8, p. 1235, 2020.

[29] V. Badrinarayanan, A. Handa, and R. S. Cipolla, “A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labeling,” https://arxiv.org/abs/1505.07293.

[30] T. Tran, O. Kwon, K. Kwon, S. Lee, and K. Kang, “Blood cell images segmentation using deep learning semantic segmentation,” in *Proceedings of the 2018 IEEE International Conference on Electronics and Communication Engineering , (ICECE)*, pp. 13–16, Xi’an, China, December 2018.

[31] H. Li, W. S. Zheng, and J. Zhang, “Deep CNNs for HEp-2 Cells classification: a cross-specimen analysis,” https://arxiv.org/abs/1604.05816.

[32] D. Yang, G. Subramanian, J. Duan et al., “A portable image-based cytometer for rapid malaria detection and quantification,” *PLoS One*, vol. 12, Article ID e0179161, 2017.

[33] K. Chakraborty, “A combined algorithm for malaria detection from thick smear blood slides,” *Journal of Health and Medical Informatics*, vol. 6, no. 1, pp. 179–186, 2015.

[34] M. Elter, E. Hasslmeyer, and T. Zerfass, “Detection of malaria parasites in thick blood films,” in *Proceedings of the 2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 5140–5144, Boston, MA, USA, December 2011.