NAME: NJOKU CHIDIEBERE CHIDUBEM

THE TASK- Predicting malaria parasite Using Computer Vision with CNN, VGG16 and RESNET50.

# 1.0 ABSTRACT

Malaria has always been a major health challenge, particularly in tropical and subtropical regions of the world (Chakradeo et al., 2021). Early detection is important to the early treatment and control of the disease. This project aims to develop an algorithm for malaria parasite prediction using computer vision via deep learning models (CNN, RESNET50 and VGG16). The models showed excellent accuracy and robust performance after rigorous testing and experimentation, suggesting that they could be used to detect malaria effectively and reliably, thereby improving healthcare outcomes in impacted areas.

# 2.0 INTRODUCTION

Malaria, a potentially lethal disease that is spread by the Plasmodium parasite, is still a major cause of death worldwide in tropical areas. This illness, in accordance to World Health Organization (WHO), affects millions of people each year.(Hemachandran et al., 2023). Early and precise parasite detection in blood samples is essential for the efficient treatment and management of malaria., hence the major motive behind the computer vision algorithm techniques for its early and accurate detection. The motive behind this project is to build an algorithm that can detect the malaria samples’ classes whether parasitic or uninfected early using different deep-learning models. This project will contribute immensely to the health sector by improving the efficiency and accuracy of early malaria detection using computer vision.

In this project, an image dataset of parasitic and uninfected malaria was already sourced. Training, validation, and testing of the dataset were carried out to ensure the generalization of the produced model. Then the trained models will undergo evaluation using performance measuring metrics like accuracy, precision, f1-score, and recall. Our respective performances will be compared to know the deep-learning technique with the best performance.

The outcome of this project will transform the high death rate associated with malaria over the years in some parts of the world and assist in early detection and diagnosis of the illness.

# 3.0 BACKGROUND

Over the years, some related works have been done on malaria parasite classifications. These works have been briefly outlined as follows:

In Malaria prediction system using deep neural network and genetic algorithm, they proposed a combination of deep learning technique (ResNet50) and a genetic algorithm optimization which yields an improved accuracy of 98.5% in malaria classification. In comparison, my project leveraged on different deep neural network performances to choose from, while the latter relies on one.

(Chakradeo et al., 2021) implemented malaria parasite detection using deep learning methods. The model is based on a VGG-type customized neural network, profiled with VGG-16 and VGG-19 networks, with two dense layers that are fully connected, and six convolutional layers and the researchers argued that the greater amount of layers does not increase the accuracy if applied to thin smear images. Cross-validation, accuracy, precision, sensitivity, specificity, and F1-score were used as the performance metrics, and it was discovered that malaria can be detected from the blood smear images giving an accuracy of about 99.3% (Shao, 1993). My project will go further in implementing more deep-learning technique to have more options to choose from.

Comparison to our study, (Chollet, 2016)used similar augmentation techniques to increase the dataset, and another similarity that may be seen in our studies is the shared goal of using deep learning techniques for diagnosing malaria, their appreciation of the value of image analysis, and their inclusion of well-known deep learning architectures in their research. However, there are still some striking differences between my project from that of theirs, which lies in their focus being the application of a unique VGG-type neural network and assessing its effectiveness on two different datasets, while my project, in contrast, examines three well- known deep learning architectures (CNN, ResNet50, and VGG16) specifically for the prediction of malaria parasites using one dataset on the three different deep neural network.

Zhaoui et al proposed using 16layer-CNN technique for its malaria classification with an average accuracy of 97.37%. we used same concept in building our CNN model from scratch.

In transfer-learning with ResNet-50 for malaria image classification, a 2layer RESnet-50 model was proposed with sigmoid activation and SGD optimizer. The performance metric measured both training and validation data with accuracies of 95.91% and 95.44% respectively. This is one of the neural network’s models implemented in this project.

# OBJECTIVES

The project objective is to reliably classify malaria images into the un-infected and the malaria parasite and compare the different neural network which is made up of convolutional neural network, which is our baseline model, ResNet50, and VGG16 using the following evaluation measuring metrics.

* + 1. Accuracy
    2. Recall
    3. F1-Score
    4. Precision

This analysis seeks to pinpoint the highs and lows of each model and offer insights into how well they can forecast malaria parasites. By attaining these goals, the research hopes to advance our knowledge of the efficiency, trade-offs, and practical considerations of applying CNN (baseline model), ResNet50, and VGG16 models for malaria parasite prediction.

# METHODOLOGY

This section involves steps in the prediction of malaria using deep learning techniques. The various methods include data collection, Image preprocessing, feature extraction, model training of CNN, Resnet50 and VGG16, and classification as shown in fig 1.



Image Preprocessing

Data Collection



Feature Extraction



Image classification

Model Training

**Fig. 1.** Flowchart diagram of the methodology used in this project.

# Data collection

The data collection for the model used was sourced from <https://www.kaggle.com/datasets/iarunava/cell-images-for-detecting-malaria> which contained the uninfected and parasitic samples. The dataset was split into training sets, validation sets, and testing sets to help in evaluating and enhancing the model's performance, assuring its generalizability to new data, and providing accurate performance metrics.

# Image Preprocessing

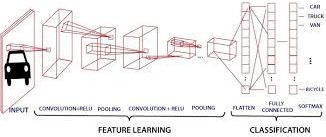
The next step was image preprocessing to guarantee that the input dimensions for the models are uniform, the images were resized to a constant size (for example, 50x50 pixels) to improve model convergence, and the pixel values normalized to a reasonable range (for example, 0-1).

# Feature Extraction

A further step taken was feature extraction, which was carried out to ensure that significant features were collected from the input images for image analysis which are later fed into fully connected layers for image classification into uninfected and malaria parasites, as the training process goes on the model optimizes its parameter for optimal image classification.

* 1. **Model Training**
     1. **CNN BASELINE MODEL (Training)**

The most popular and commonly applied deep learning algorithm is CNN. The fact that CNN completes the operation without human assistance gives it a significant advantage over its forerunners. automatically recognizing the crucial elements (Taye, 2023). The structures seen in both animal and human brains, classical neural networks, are the inspiration for CNN modelling. CNN uses shared weights and local connections to fully exploit 2D input-data structures, such as visual signals, in contrast to conventional fully connected (FC) networks (Alzubaidi et al., 2021). This method streamlines training and speeds up the network by utilizing a small set of parameters. Cells in the visual cortex also exhibit this. It's important to note that just a little portion of a scene can be detected by these cells (Alzubaidi et al., 2021).



**Fig. 2**. CNN architecture of image classification (Saturn Cloud Blog, 2023)

A CNN architecture for categorizing images is shown in an example in Fig 2. A CNN model's inputs come in three dimensions: height (m), breadth (m), and depth (m), with height (m) being equal to the width. The channel number can also be referred as depth name for depth. An RGB image, for instance, has a depth (r) of three. Using the train test split () function from the scikit- learn module, the dataset will be split into training and test sets. Then, using element-wise division, the training and test sets will be normalized to a maximum value of 1. The matplotlib package will then be used to render the images. The CNN model will be created using the Keras package. This model consists of six convolutional layers, each followed by a relu activation function and a max-pooling layer, with progressively more filters. After being flattened, two dense layers with relu activation functions will be added to the output of the final max-pooling layer.

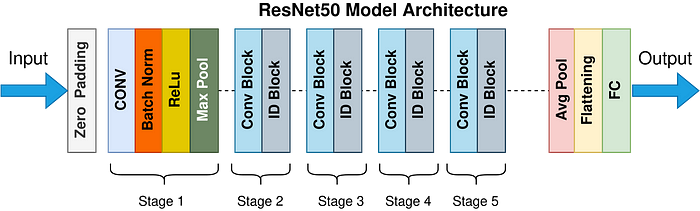
The model will become nonlinear because of the activation layers and the activation function

of the final dense layer will be sigmoid which will classify the malaria cells into classes.

* + 1. **ResNet50 MODEL (Training)**

A ResNet50 transfer learning model is the second model that needs to be trained. The ResNet50 model will be loaded with previously trained weights and frozen to prevent alterations during training (Gupta et al., 2022). The pre-trained ResNet50 model will be used as a feature extractor, and a few custom layers will be added to the group the malaria cells into classes.

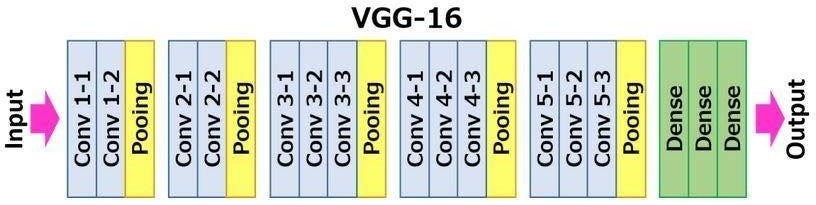
The vanishing gradient issue in extremely deep networks is addressed by ResNet50, a deep residual network architecture. It introduces the idea of residual connections, often known as skip connections, which allow data to flow through some layers without being affected (Simonyan & Zisserman, 2015). The residual connections aid the model to learn residual mapping ensuring that the models are optimized.



**Fig. 3.** RESNet50 architecture of image classification (Bendjillali et al., 2020).

* + 1. **VGG16 MODEL (Training)**

A VGG16 transfer learning model will be the third model to be trained. VGG16 is well known for its optimal performance on the ImageNet dataset. The VGG16 model will be frozen and loaded with pre-trained weights to prevent any changes during training. To divide the malaria cells into classes, a few special layers will be added to the pre-trained VGG16 model, which will be used as a feature extractor.



**Fig. 4.** VGG16 architecture of an image classification (Rohini, 2021).

These three architectures offer various methods for feature extraction and representation learning, which may have an impact on how well the models classify malaria parasites. In the end, each model's accuracy, confusion matrix, and classification report will be compared to determine which model performed the best. The most effective model will be recommended.

# EXPERIMENTS

* 1. **Sourcing of data**

This project has been sequential, as the first stage was the data collection of 27,558 images. This project's dataset was obtained via the open source Kaggle website. The dataset can be found at this link: [https://www.kaggle.com/datasets/iarunava/cell-images-for-detecting-](https://www.kaggle.com/datasets/iarunava/cell-images-for-detecting-malaria) [malaria.](https://www.kaggle.com/datasets/iarunava/cell-images-for-detecting-malaria)

# Image Preprocessing

The next step taken was image preprocessing, we used data\_dir = "./Malaria\_images/" to read our images, and then the images were then resized to batch size = 50 images size = (50, 50,3), where 50pixels= height, 50pixels= width and 3 = channels, showing that it is a colored image. Having resized all our images to the same size, we now fed them into our deep learning models.

# Input(X) and output(y) assigning.

The next step in the procedure involved creating the input properties and labels for the model. The photos that had been scaled were kept as the input features X in a NumPy array. To make sure that X's form matched the model's anticipated input measurements, it was printed in that shape. Additionally, labels were created for every image. To create labels for that directory, a for loop iterating over the photographs in the directory. A list of integers indicating the image categories made up of the labels that were generated. X.shape generated was (27558,50,50,3), while the shape was (27558).

# Conversion of Lists to NumPy arrays

The next step was to convert the list to NumPy arrays to ensure interoperability with deep learning frameworks and libraries, reduce the complexity associated with data handling, and increase computational effectiveness.

# .5 Splitting of the dataset into training, validation, and testing sets

I went further to split the dataset into training, validation, and testing. The training set of 17636 images was generated and used to train the deep learning model to learn patterns and relationships in the data. The validation set of 4410 images was generated and used to make decisions about the model hyperparameters for optimal performance. A testing set of 5512

(20%) images was generated and plays a critical role in determining how well a deep learning model performs in the real world and offers insightful information about its generalization potential and prediction accuracy using unknown data.

# conversion into label encoder and augmentation of the data set

Categorical labels or classes are transformed into numerical values through the label encoding procedure. So, label encoding was further carried out on the dataset to encode the target variable or class labels(y) in a way that the model can comprehend and interpret. Further data augmentation like random zoom, random flip and random rotation was carried out on our dataset for higher performance, better generalization, and increased robustness when dealing with real-world scenarios by learning from a more varied and representative dataset.

# Model Construction

This stage involves the building of three models: the CNN model, ResNet50, and VGG16. I started with the sequential architecture in Keras for image classification, then with data augmentation to improve generalization, Conv2D layer with 16 filters, (2, 2) kernel size, and "relu" activation function are the model's initial building blocks. On the input pictures, this layer applies convolutional techniques and an activation function to add non-linearity. The next layer is the MaxPooling2D layer, which has a pool size of (2, 2). Max pooling is used to keep the most noticeable characteristics while reducing the spatial dimensions of the feature maps. The following next two Conv2D layers have 32 and 64 filters, respectively. These layers take the input data's features and extract them further.

Then I implored the flattened layer which transforms the multidimensional feature maps into a 1D vector. The convolutional layers must be connected to the fully connected layers via this flattening step. Then dropout of 0.2 was applied twice to avoid overfitting. Then two fully connected layers with the first layer being 500 and using the relu activation function, the second dense layer utilizes the ‘sigmoid' activation function and has one node. A value close to 1 denotes the presence of one class, such as malaria, while a value close to 0 denotes the absence of that class, such as malaria. This final layer generates the output probabilities for binary classification. The categorical cross-entropy loss function was minimized using the Adam) method. The Adam optimizer's learning rate was set to default, batch size of 50 and the model was trained for 20 epochs. These hyperparameters were utilized for training.

The pre-trained weights of ResNet50 and VGG16 were leveraged via transfer learning. A new dense layer with a sigmoid activation function was created after the old dense layer was removed. Following the final convolutional block's adjustment, the models were trained using the same hyperparameters as the conventional CNN model.

Finally, the evaluation performance metrics were now implemented to analyze how effective our respective models were. The metrics used were accuracy, F1-score, Precision, and recall.

# 7.0 RESULTS

Having implemented several stages ranging from data collection to classification, the three models were able to classify the images correctly. Performance evaluation metrics were used to compare the performances of the three models. The table below show the classification report for each of the models.

**Table1.** Performance metrics of our three classification models.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Performance metrics** | **Accuracy** | **Recall** | **F1score** | **precision** |
| CNN | 0.96 | 0.96 | 0.96 | 0.96 |
| VGG16 | 0.91 | 0.90 | 0.90 | 0.91 |
| RESNET-50 | 0.70 | 0.70 | 0.69 | 0.72 |

For assessing classification models, the following metrics were employed:

# Accuracy

For both parasite and uninfected images, accuracy gauges how accurately the algorithm has classified the outcomes overall. It determines the proportion of accurately classified samples (both parasitic and uninfected samples) to all the samples in the dataset.

# Precision:

Precision in this context refers to the percentage of parasitic images that were accurately predicted out of all images that were anticipated to be parasitic. It focuses on the model's capacity to prevent false positives, evaluating how well the model can distinguish between genuine parasitic photos and false parasitic images that are wrongly labelled as non-parasitic.

# Recall

This is the percentage of parasitic images that were accurately predicted out of all actual parasitic images. It evaluates the model's capacity to identify every parasitic occurrence without missing out any. A higher recall shows that the model successfully recognizes the majority of the real parasite photos.

# F1-Score

It offers a statistic that combines precision and recall trade-offs into a single value. A higher F1 score denotes a better equilibrium between accurately identifying parasitic images (precision) and recalling the bulk of the actual parasite images.

From the above evaluation performance tables and in terms of model comparison, it is interesting to note that our three models performed well, and it is glaring that our baseline model, which is CNN performed the most, with the highest evaluation performance metrics. This is likely because it successfully captures the dataset's fundamental properties without running the danger of overfitting or necessitating an overly complex model. This approach effectively captures the key features of the dataset while avoiding the risks of overfitting or requiring a complicated model.

# 8.0 RECOMMENDATION

Future uses of this study include expanding the dataset to encompass a wider range of scenarios and test various deep learning architectures to enhance classification performance and methods like attention mechanisms and Grad-CAM can be used to enhance the model's interpretability.

# 9.0 CONCLUSION

In conclusion, this study suggested the classification of malaria parasites using deep learning architectures CNN, VGG16, and ResNet50. Our study demonstrated how well these models performed in terms of performance evaluation metrics with CNN(baseline) achieving highest accuracy of 96%, followed by VGG16 achieving 91%, and ResNet50 attaining 70% accuracy on a broad dataset of malaria parasites. The results highlight how deep learning could improve early malaria detection and classification to minimize the death mortality associated with this disease.

# 7.0. REFERENCES

*A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way | Saturn Cloud Blog* (no date). Available online: https://saturncloud.io/blog/a-comprehensive-guide-to-convolutional-neural-networks- the-eli5-way/ [Accessed 02/07/2023].

Alzubaidi, L., Zhang, J., Humaidi, A.J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M.A., Al-Amidie, M. & Farhan, L. (2021) Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data 2021 8:1*, 8(1), 1–74. Available online:

https://doi.org/10.1186/S40537-021-00444-8.

Bendjillali, R.I., Beladgham, M., Merit, K. & Taleb-Ahmed, A. (2020) Illumination-robust face recognition based on deep convolutional neural networks architectures. *Indonesian Journal of Electrical Engineering and Computer Science*, 18(2), 1015–1027. Available online:

https://doi.org/10.11591/IJEECS.V18.I2.PP1015-1027.

*Building powerful image classification models using very little data* (no date). Available online: https://blog.keras.io/building-powerful-image-classification-models-using-very-little-data.html [Accessed 02/07/2023].

Chakradeo, K., Delves, M., of, S.T.-I.J. & 2021, undefined (no date) Malaria parasite detection using deep learning methods. *Researchgate.Net* [Preprint]. Available online:

https://doi.org/10.5281/zenodo.4569849.

*Everything you need to know about VGG16 | by Great Learning | Medium* (no date). Available online: [https://medium.com/@mygreatlearning/everything-you-need-to-know-about-vgg16-7315defb5918](https://medium.com/%40mygreatlearning/everything-you-need-to-know-about-vgg16-7315defb5918) [Accessed 02/07/2023].

Gupta, P.R., Sharma, D. & Goel, N. (2022) Image Forgery Detection by CNN and Pretrained VGG16 Model., 141–152. Available online: https://doi.org/10.1007/978-981-16-6887-6\_13.

Hemachandran, K., Alasiry, A., Marzougui, M., Ganie, S.M., Pise, A.A., Alouane, M.T.H. & Chola, C. (2023) Performance Analysis of Deep Learning Algorithms in Diagnosis of Malaria Disease. *Diagnostics 2023, Vol. 13, Page 534*, 13(3), 534. Available online: https://doi.org/10.3390/DIAGNOSTICS13030534.

Shao, J. (1993) Linear Model Selection by Cross-Validation. *Journal of the American Statistical Association*, 88(422), 486. Available online: https://doi.org/10.2307/2290328.

Simonyan, K. & Zisserman, A. (2015) Very deep convolutional networks for large-scale image recognition.

*3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*

[Preprint].

Zhaoui Liang., Andrew Powell., IlkerErsoy.,Mahdieh Poostchi., Kamolraut Silmaut and Kannapan Palani.(2017) *CNN-based image analysis for malaria diagnosis” IEEE International*

*Conference on Bioinformatics and Biomedicine (2017).*

Taye, M.M. (2023) Theoretical Understanding of Convolutional Neural Network: Concepts, Architectures, Applications, Future Directions. *Computation 2023, Vol. 11, Page 52*, 11(3), 52. Available online: https://doi.org/10.3390/COMPUTATION11030052.