Optimized Deep Learning Model for Early Detection and Classification of Lung Cancer on CT Images

UNIVERSITY OF
KWAZULU - NATAL

INYUVESI
YAKWAZULU-NATALI

Tehnan Ibrahem Alhasseen Mohamed

February, 2022

Optimized Deep Learning Model for Early Detection and Classification of Lung Cancer on CT Images

by

Tehnan Ibrahem Alhasseen Mohamed

A thesis submitted to the
University of KwaZulu-Natal
in fulfilment of the requirements for the degree
of
MASTER OF SCIENCE
in
COMPUTER SCIENCE

Thesis Supervisor: Dr. Absalom El-Shamir Ezugwu

Thesis Co-supervisor: Dr. Olaide Nathaniel Oyelade

UNIVERSITY OF KWAZULU-NATAL

SCHOOL OF MATHEMATICS, STATISTICS AND COMPUTER SCIENCE

PIETERMARITZBURG CAMPUS, SOUTH AFRICA

Declaration - Plagiarism

I, Tehnan Ibrahem Alhasseen Mohamed, declare that

1. The research reported in this thesis, except where otherwise indicated, is my original research.
2. This thesis has not been submitted for any degree or examination at any other university.
3. This thesis does not contain other persons' data, pictures, graphs or other information, unless specifically acknowlegded as being sourced from other persons.
4. This thesis does not contain other persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then
   (a) their words have been re-written but the general information attributed to them has been referenced, or
   (b) where their exact words have been used, then their writing has been placed in italics and referenced.
5. This thesis does not contain text, graphics or tables copied and pasted from the internet, unless specifically acknowledged, and the source being detailed in the thesis and in the reference sections.

Tehnan Ibrahem Álhasseen Mohamed (Student) Date 22-03-2022

2022/03/22

Dr. Absalom El-Shamir Ezugwu (Supervisor) Date

2022/03/22

Dr. Olaide Nathaniel Oyelade (Co-supervisor) Date

Disclaimer

This document describes work undertaken as a Masters programme of study at the University of KwaZulu-Natal (UKZN). All views and opinions expressed therein remain the sole responsibility of the author, and do not necessarily represent those of the institution.

Abstract

Recently, researchers have shown an increased interest in the early diagnosis and detection of lung cancer using the characteristics of computed tomography (CT) images. The accurate classification of lung cancer assists the physician to know the targeted treatment, reducing mortality, and as a result, supporting human survival. Several studies have been carried out on lung cancer detection using a convolutional neural network (CNN) models. However, it still remains a challenge to improve the model's performance. Moreover, CNN models have some limitations that affect their performance, including choosing the optimal architecture, selecting suitable model parameters, and picking the best parameter values for weights and bias. To address the problem of selecting the best combination of weights and bias needed for the classification of lung cancer in CT images, this study proposes a hybrid of Ebola optimization search algorithm (EOSA) and the CNN model. We proposed a hybrid deep learning model with preprocessing features for lung cancer classification using publicly accessible Iraq-Oncology Teaching Hospital/National Center for Cancer Diseases (IQ-OTH/NCCD) dataset. The proposed EOSA-CNN hybrid model was trained using 80% of the cases to obtain the optimal configuration, while the remaining 20% was applied for validation. Also, we compared the proposed model with similar five hybrid algorithms and the traditional CNN. The results indicated that EOSA-CNN scored 0.9321 classification accuracy. Furthermore, the result showed that EOSA-CNN achieved a specificity of 0.7941, 0.97951, 0.9328, and sensitivity of 0.9038, 0.13333, 0.9071 for normal, benign, and malignant cases, respectively. This confirmed that the hybrid algorithm provides a good solution for the classification of lung cancer.

Dedication

This dissertation is dedicated to
my supervisors,
my parents,
my brothers,
my sisters,
teachers,
and friends
who have always taught me how to see the vitality in myself,
conquer fears and overcome challenges.
I hope that I have made you proud.

Acknowledgements

In the name of Allah, most Gracious, most Merciful. Praise be to God, the Cherisher and Sustainer of the world. I am highly grateful to the Almighty Allah for His unmeasurable Mercies that saw me through my studies. It is such a great honour for me and humbling at the same time because it has been a long journey characterized by long hours of hard work and sacrifice. It is with pleasure and gratitude to acknowledge those who rendered assistance through the lengthy process of my study. I sincerely appreciate all the big and small contributions from everybody who either directly or indirectly helped make my studies less stressful and a success.

I am specifically grateful to my mentors and supervisors, Dr. Absalom El-Shamir Ezugwu and Dr. Olaide Nathaniel Oyelade for their presence, inspiration, patience, dedication and their powerful guidance and help. I again thank them for their availability and readiness to read and correct all the many errors in my work, as well as the financial support. Much thanks for their quick responses whenever I consulted. Their help in and outside the research has been immense. I am deeply grateful for their willingness to work with me. Thank you both for sharing your enormous statistical knowledge with me. I would like to express my profound gratitude to the School of Mathematics, Statistics, and Computer Science, College of Agriculture, Engineering, and Science, University of KwaZulu-Natal, South Africa for the support and financial help to complete this Masters project successfully. I would also like to thank Miss Christel Bernard for ensuring I have a comfortable working place environment. I would also like to thank Mr. Osama Salim who helped me in the language editing.

I would like to thank my parents, Ibrahem and Fatheya who have taught me the important things in life and have supported all my endeavours, for their unconditional love, encouragement, patience for being away when they needed me the most. I am grateful and thankful to sisters, Nehad, and Emtenan for their caring of my parents. Many thanks to the rest of my extended family for their moral support.

Above all I would like to thank my husband Mohanad Mohammed for his love and constant support, for all the late nights and early mornings, and for keeping me sane over the past few months. Thank you for being my muse, editor, proof-reader, and sounding board. But most of all, thank you for being my best friend. I owe you everything. Last, but not least, thanks to everyone who his or her names may not appear here, but I won't forget their help. May the Almighty Allah bless them all.

Derived Publications

The following publications are derived from this study:

1. **Tehnan I. A. Mohamed**, Olaide N. Oyelade, Absalom E. Ezugwu, Laith Abualigah, (2022). Ebola Optimization Search Algorithm with Application to Deep Learning Model for Early Detection and Classification of Lung Cancer on Computed Tomography Images. Computer in Biology and Medicine. (**Under review: The manuscript submitted to Neural Computing Applications journal, springer**)

2. O. N. Oyelade, A. E. Ezugwu, **T. I. A. Mohamed** and L. Abualigah, "Ebola Optimization Search Algorithm: A new nature-inspired metaheuristic algorithm with application in medical image classification problem," in IEEE Access, doi: 10.1109/ACCESS.2022.3147821.

Contents

List of Figures

List of Tables

Abbreviations

CT	Computed Tomography
CNN	Convolutional Neural Network
IQ-OTH/NCCD	Iraq-Oncology Teaching Hospital/National Center for Cancer Diseases
EOSA	Ebola Optimization Search Algorithm
NCDs	Non-Communicable Diseases
WHO	World Health Organization
SCLC	Small Cell Lung Cancer
NSCLC	Non-Small Cell Lung Cancer
HIV	Human Immunodeficiency Virus
DL	Deep Learning
ANNs	Artificial Neural Networks
GA	Genetic Algorithm
WOA	Whale Optimization Algorithm
MVO	Multiverse Optimizer
SBO	Satin Bower Optimization
LCBO	Life Choice-Based Optimization
SVM	Support Vector Machine
NNs	Neural Networks
KASC	Kernel Attribute Selected Classifier
FFBPNN	Feed-Forward Back Propagation Neural Network
Grad-CAM	Gradient Weighted Class Activation Mapping
FDG-PET/CT	fluorodeoxyglucose positron emission tomography/computed tomography
LIDC	Lung Image Database Consortium
LUNA16	Lung Nodule Analysis 2016
PSGIMSR	PSG Institute of medical sciences and research
DITNN	Deep Learning wiht Instantaneously Trained Neural Networks
R2MNet	Radiology Analysis and Malignancy Evaluation Network
CDAM	Channel-Dependent Activation Mapping
DNN	Deep Neural Network
AUC	Area Under Curve
MIL	Multiple Instance Learning
SEM	Standard Error of the Mean
MAN	Modified AlexNet
MLP	Multi-Layer Perceptron
CADx	Computer Aided Diagnosis
DBN	Deep Belief Network
SDAE	Stacking Denosing Autoencoder

ALCD	Adoptive Lung Cancer Detection
SIFT	Scale Invariant Feature Transform
CAD	Computer Aided Detection Systems
IHC	Immunohistochemistry
HOG	Histogram of Oriented Gradient
FPSO	Fuzzy Particle Swarm Optimization
GCPSO	Guaranteed Convergence Particle Swarm Optimization
ITEO	Improved Thermal Exchange Optimization
DWT	Discrete Wavelet Transform
CLAHE	Contrast-Limited Adaptive Histogram Equalization
CR	Corner region
BR	Border Region
IR	Inner Region
CC	Current Chance
TP	True Positive
FP	False Positive
FN	False Negative
TN	True Negative
SD	Standard Deviation
LL	Low-Low
HL	High-Low
LH	Low-High
HH	High-High
WEP	Wormhole Existence Probability
TDR	Traveling Distance Rate
EVD	Ebola Virus Disease

Chapter 1

Introduction

Non-communicable diseases (NCDs) are considered as the major causes of death globally. The NCDs include several types of diseases such as cancers, cardiovascular diseases, chronic respiratory diseases, and type-two diabetes Caprara (2021). Moreover, the World Health Organization (WHO) estimates that NCDs are responsible for 80% of the world's burden of disease in 2020 and represent 71% of the worldwide deaths Wang & Wang (2020); Bigna & Noubiap (2019). In addition, the global burden of NCDs is expected to increase by 17% in the coming decade Wang & Wang (2020). Recently public health investigation revealed an epidemiological shift from infectious to NCDs. Among the NCDs, cancer has the highest morbidity and mortality rates worldwide. The incidence and spread of cancer are on the rise globally, both in developed and developing countries Olsen (2015).

Cancer is a severe public health issue that is becoming more prevalent worldwide. It is a disease that makes cells tissues in particular divide in uncontrolled way. Therefore, this leads to a malignant or tumor growth Woodman et al. (2021). In 2020, the GLOBOCAN estimated 19.3 million new cancer incidences that led to approximately 10 million deaths globally Sung et al. (2021). Also, the American Cancer Society estimated 1,898,160 new incidences of cancer that will lead to 608,570 deaths in the United States (US) Siegel et al. (2021). Moreover, in 2020 there were 1,109,209 new cases and about 711,429 deaths in Africa, even though the cancer death rates exceed the combined death rate of AIDS, tuberculosis, and malaria Hamdi et al. (2021); Globocan (2021).

Lung cancer is one of the most often diagnosed cancer, and it is the main cause of death for men and women globally. There are 2.2 million new cases of lung cancer that have been diagnosed, which will lead to approximately 1.8 million death globally Sung et al. (2021); Mapanga et al. (2021). There are several common signs and

symptoms of lung cancer, which include: hemoptysis (coughing up blood), weight loss, and weariness. Moreover, various risk factors are associated with lung cancer, including smoking, alcohol consumption, air quality, and food Li et al. (2021). Lung cancer can be divided into two categories based on the histology of the cancer cells: non-small lung cancer (NSCLC) and small-cell lung cancer (SCLC) Woodman et al. (2021). The NSCLC is considered as the most common type of lung cancer, which accounts for 85% compared to the SCLC, which represents 15% of all patients Woodman et al. (2021).

Lung cancer has significantly increased in the third world countries, including Sub-Saharan Africa, where human immunodeficiency virus (HIV) has also had a devastating effect Shankar et al. (2019). The overall 5-year survival rate for all kinds of lung cancer is lower (18%) when compared to other cancers such as colorectal cancer (65%), breast cancer (90%), and prostate cancer (99%) Woodman et al. (2021). However, lung cancer demands greater attention from the medical, biological, and scientific fields to find innovative solutions to promote early diagnosis, help in medical decisions, and evaluate responses to improve health care. The molecular profile of tumor tissues allows for identifying driver mutations, and tailored therapies that can be applied for particular genotypes. Traditional chemotherapy kills all cells without distinguishing between normal and malignant ones. on the other hand, targeted therapy targets specific parts, interacting with cancer driver genes and preventing or decreasing malignant transformation Pereira et al. (2021).

A significant proportion of the cancer incidences that are curable in developed countries are detected after they reach incurable stages in third world countries due to late or inaccurate diagnoses Organization (2002). This has motivated scientist to evaluate the existing approaches and propose new techniques to classify and detect lung cancer and its subtypes for improved early detection and application of appropriate treatment strategies.

There is an enormous amount of CT scan images data repository for lung cancer, which could help detect the disease. More so, machine learning and deep leaning algorithms can utilize these images to enhance cancer prediction and diagnosis as early as possible and find the best treatment strategies. There are two types of machine learning methods: supervised and unsupervised. The supervised algorithms use labeled input and output data (for example, labeled pictures of malignant and benign scans). In contrast, the unsupervised algorithm aims to find the undiscovered relationships and patterns within the data without explicitly labeling. The main aim of using machine and deep learning is usually to learn how to categorize an input

sample (such as medical, MRI scans, or pathological images) into predetermined categories (such as normal or abnormal). These categories could be specific sub-type diagnoses (for example, tumors) or diagnostic groups (e.g., malignant versus benign) Daneshjou et al. (2021). Deep Learning (DL) methods have enabled machines to analyze high-dimensional data such as images, multidimensional anatomical images, and videos. Also, the DL can be considered as a branch of machine learning that explains learning algorithms inspired by the biological and function of the nervous system. DL uses a combination of Artificial Neural Networks (ANNs) to improve recognition skills Masud et al. (2021).

Optimization is a critical phenomena in deep learning Sun (2020). It can be defined as a process of training the model iteratively, which results in a maximum and minimum function evaluation. Deep learning algorithms have been applied in various fields successfully. However, optimizing the parameters of these algorithms has become a complex task due to the enormous amount of data required to successfully conduct the required training and validation of the models Soydaner (2020). Thus, there is a need for improved methods of optimizing the deep learning hyperparameters using selected state-of-the-arts optimization algorithms with application to enhanced image classification.

1.1 Motivation

Despite the higher mortality rate in Africa, cancer has received little attention from researchers and health providers. Furthermore, low-income nations account for 57% of all new cancer cases worldwide, which increased by a lack of awareness, low protective initiatives, and increased life expectancy Hamdi et al. (2021).

Lung cancer is the most common type of cancers, and it is connected with high morbidity and mortality rates, especially if detected at a late stage Mapanga et al. (2021). Also, there is a lack of methods that can effectively detect lung cancer in its early stages, where it is more likely to be curable. Moreover, the symptoms of lung cancer usually manifest late when it is non-curable. Although early detection is the best promising way to reduce any health complications or death due to cancer, more so, 80% of the cases are detected correctly in the middle or advanced stage of cancer Motohiro et al. (2002); Kuruvilla & Gunavathi (2014).

Furthermore, there are severe limitations in cancer detection based on morphological features Berman (2004). The early diagnosis and detection of lung cancer via CT images will help the radiologists make a fast decision, hence expediting and facil-

itating the consequent clinical response to determine the suitable treatment. Thus, these challenges motivated the current study proposal to develop effective DL methods for early diagnosis and detection of lung cancer based on CT images to enhance the survival chances of humans.

1.2 Problem Statement

Although several studies have reported various designs of CNN models developed for medical images analyses, lung cancer prediction still remains a difficult task due to the multifaceted designs in the CT scan. Moreover, DL models have some issues that affect their performance, including choosing the feature representation, the optimal architecture, selecting suitable model parameters, and picking the best combination of values for weights and bias Akay et al. (2021). Therefore, to solve these issues and find a precise prediction model, we used metaheuristic optimization methods to optimize the CNN model. Metaheuristic optimization approaches helps to optimize the performance of CNN architecture network model. Thus, we propose utilizing a metaheuristic algorithm namely Ebola optimization search algorithm (EOSA) to address the problem at hand. The use of EOSA is motivated from the interesting performance reported in applying the method to CNN auto-design for image classification problem in breast cancer Oyelade & Ezugwu (2021a). The CT images have noises at the time of acquisition process which might lead to false detection of the cancerous lung. Moreover, good image preprocessing, such as wavelet decomposition, will be used to enhance image resolution. In addition, detecting lung cancer is still challenging for physicians as it is difficult to conduct such investigation using eye-tracking and then recognize the lung nodule and their sub-types.

1.3 Objectives

The main objective of this work is to design an optimized deep learning model using a metaheuristic algorithm for lung cancer detection. Hence, this will help the physicians in the early detection of the disease, and thereby improving decision making leading to proper therapy. The primary objectives of this thesis are summarized as follows:

1. Apply combined transformation, Gaussian Blur filter, otsu's thresholding, normalization, erosion and dilation, CLAHE filter, and wavelet decomposition, for preprocessing of the CT images related to cancer diagnosis.

2. Design and implement a robust hybrid deep learning model that combines both the CNN and EOSA, to optimize the CNN architecture.

3. Evaluate and compare the new deep learning model with other state-of-the-art hybrid algorithms such as the combination of CNN with genetic algorithm (GA), whale optimization algorithm (WOA), multiverse optimizer (MVO), satin bower optimization (SBO), and life choice-based optimization (LCBO).

1.4 The Structure of the Dissertation

This work proposes an optimized deep learning approach that will help in the early diagnosis and detection of lung cancer via CT scan images. The dissertation consists of five chapters. These chapters are structured as follows:
Chapter 1: This chapter gives a background about lung cancer and deep learning models. Also it presents the motivation, aims, and objectives of the study.
Chapter 2: This chapter presents the literature review for the studies which are recent and related to our research.
Chapter 3: This chapter describes the dataset for lung cancer and presents a detailed preprocessing approach as applied in this study. Furthermore, we introduced the proposed hybrid algorithm and similar CNN-metaheuristic methods such as: CNN using GA, WOA, MVO, SBO, and LCBO for early detection and diagnosis of lung cancer. The methodology that are applied to detect and diagnose lung cancer are also explained.
Chapter 4: Here, we present results for seven methods applied to the dataset. The results are compared for the different methods based on various performance measures, namely accuracy, sensitivity, specificity, kappa, recall, precision, and f1 score.
Chapter 5: In this chapter, we provided a discussion about the outcome of the proposed methods and results obtained. We also concluded the study and presented future research directions.

Chapter 2

Literature Review

2.1 Introduction

Recently, research efforts have demonstrated the existence of an increased spread of NCDs such as cancer. Lung cancer diagnosis and detection have become one of the biggest obstacles in recent years. Early diagnosis and detection of lung cancer will reliably promote survival of many lives Kalaivani et al. (2020). Improved classification accuracy for lung cancer using medical images will help physicians select the suitable therapy that will reduce the mortality due to the disease.

Similarly, there is a growing trend using CT images, which is far more effective than X-ray, for classification of lung cancer. Moreover, several machine and deep learning classification algorithms have been implemented to address the problem of classification of lung cancer. These algorithms include CNN, recurrent neural network (RNN), support vector machine (SVM), neural networks (NNs), among others. Here, we categorized the literature review into three parts based on different approaches that have been investigated and implemented.

2.2 Machine learning based lung cancer diagnosis

Several computer-aided methods have investigated pre-processing and machine learning methods to lung cancer diagnosis. Makaju et. al. Makaju et al. (2018) proposed a new computer-aided system for cancerous nodule classification based on CT scan images. The novel model uses watershed segmentation for detection and SVM to classify a nodule as either malignant or benign. Their results showed that the new proposed model accurately classifies cancer with 92%. Also, Nanglia et. al. Nanglia et al. (2021) presented a hybrid algorithm called kernel attribute selected classifier (KASC) based on combining SVM and feed-forward back propagation neural net-

work (FFBPNN). They evaluated the hybrid method using CT images. The results of the new hybrid technique achieved high performance with an average accuracy of 98.17% for the classification of 500 CT images of lung cancer. In addition, Kareem et al. (2021) proposed a new model for detecting lung cancer using the IQ-OTH/NCCD CT images. They implemented preprocessing steps namely, image enhancement, segmentation, and feature extraction. Thereafter, they used SVM to classify the CT images. Their results showed an accuracy of 89.88%.

2.3 Classical CNN based lung cancer diagnosis

Many works have been done in lung cancer detection using CNN. Hatuwal & Thapa (2020) applied CNN model to detect lung cancer using histopathology images obtained from the LC25000 Lung and colon histopathological image dataset. The dataset is divided into 90% and 10% for training and validation. The results obtained showed that CNN achieved 96.11% and 97.2% for training and validation respectively. Also, Neal Joshua et al. (2021) conducted binary classification of benign and malignant categories of lung cancer using the Lung Nodule Analysis 2016 (LUNA16) database. They proposed a lightweight 3D-CNN model that uses gradient-weighted class activation mapping (Grad-CAM). The proposed model performance is higher than 2D AlexNet and 3D AlexNet with an accuracy of 97.17%.

Kirienko et al. (2018) designed a model based on deep learning for lung cancer stages detection. They used fluorodeoxyglucose positron emission tomography/computed tomography (FDG-PET/CT) images collected from 472 patients. Their developed model obtained an accuracy of 87% in training, 69% in validation, and 69% in testing. Moreover, Sajja et al. (2019) developed a model based on GoogleNet, a pre-trained CNN, for lung cancer classification using lung image database consortium (LIDC) dataset. They divided the datasets into 80% as training set and 20% as testing set. The results of the proposed model outperformed AlexNet, GoogleNet, and ResNet50 resulting in the highest accuracy.

Ahmed et al. (2020) applied the Vanilla 3D CNN model to classify cancerous or non-cancerous lung cancer. They used the LUNA 16 dataset and pre-processed the data using thresholding methods. The results of their proposed model showed an accuracy that is close to 80%. Furthermore, Khumancha et al. (2019) utilized a model based on 3D-CNN for lung cancer identification. The authors used LUNA16 and the data science bowl 2017 kaggle competition datasets. The model prediction accuracy using the data science bowl 2017 Kaggle competition dataset was impressive. The results showed that the model achieves 89.24% and 82.17% of precision and recall re-

spectively. Yamunadevi & Ranjani (2021) proposed an adaptive fuzzy-GLCM based segmentation method for lung cancer recognition. They used the GoogLeNet CNN architecture. The result of the proposed model yielded 98% accuracy, specificity, sensitivity, and precision compared to existing methods. In addition, Toğaçar et al. (2020) presented a hybrid model using six machine learning models and three CNN models. Also, they used the minimum redundancy maximum relevance (mRMR) feature selection approach to detect lung cancer via CT images. They used 10-fold cross-validation and carried out five separate experiments. Their results showed that the combination of AlexNet and KNN with mRMR method scored an accuracy, sensitivity, and specificity of 99.51%, 99.32%, 99.71%, respectively.

Cengil & Cinar (2018) applied 3D-CNN architecture on CT images to classify lung cancer nodules. They obtained their test images from SPIE-AAPM-LungX dataset. Their results showed that the proposed method scored 70% accuracy. Also, Chen et al. (2021) developed a new synthetic sample generation method based on the characteristics of multiple instance learning (MIL) to solve the problem of imbalance data. Their results achieved an average accuracy of 0.807, recall of 0.870, and AUC of 0.842. Singh & Gupta (2019) developed a method for detecting and classifying lung cancer into benign and malignant stages using CT scan images. They used image processing techniques to prepare the images for further analysis. In addition, the study used seven classification methods: decision tree classifier, multi-layer perceptron (MLP), k-nearest neighbors classifier, multinomial naive Bayes classifier, support vector machine classifier, stochastic gradient descent classifier, and random forest classifier. Their results indicated that MLP outperformed other methods with an accuracy of 88.55%.

Also, Sujitha & Seenivasagam (2021) proposed a new approach for lung cancer classification by streamlining Apache Spark designs and architecture with machine learning algorithms together. Their approach outperformed other methods and obtained an accuracy of 86% and AUC of 0.88. Alrahhal & Alqhtani (2021) presented adoptive lung cancer detection (ALCD) system using CNN model. The authors implemented preprocessing methods for preparing the images for further analysis. Moreover, the used feature extraction method is called scale-invariant feature transform (SIFT). Their results indicated that the proposed system scored 96%, 98%, 92%, and 96% for accuracy, sensitivity, recall, and precision, respectively.

Kriegsmann et al. (2020) trained and refined a CNN model to consistently classify the three most common lung cancer subtypes. They also built quality control procedures to objectively identify situations that require additional immunohistochem-

istry (IHC) validation for accurate entity subtyping. Their results achieved 88% of accuracy. Moreover, Al-Yasriy et al. (2020) introduced a computer-aided system for lung cancer detection using the IQ-OTH/NCCD dataset. They combined CNN with AlexNet architecture to classify the patients into normal, benign, or malignant. Their proposed method achived an accuracy of 93.548%.

2.4 Previous Empirical Research Studies

Revathi et al. (2020) proposed a new hybrid deep learning classifier based on adaptive swarm intelligence for classifying the lung cancer nodules. They used LIDC and real-time lung cancer data samples from PSG Institute of medical sciences and research (PSGIMSR). Their proposed algorithm aimed to improve the exploration and exploitation mechanisms of the optimal search process to find near-optimal solutions. Their proposed model showed an enhancement in lung cancer detection with mean square error rate 0.0018 and 0.0027 for LIDC and PSGIMSR respectively. Also, Das et al. (2020) proposed a velocity-enhanced whale optimization technique. Their proposed method uses a hybrid approach with an artificial neural network for detection and diagnosis of various cancer types including, lungs, cervical and breast cancer. They compared their proposed technique with four benchmark algorithms. These algorithms are: linear discriminate analysis (LDA), C4.5, factorized distribution algorithm, and learning vector quantization. Their proposed approach scored a classification accuracy of 97.65%, 94.6%, and 84% in breast, cervical, and lung cancer, respectively.

In addition, Shakeel et al. (2019) used an enhanced profuse clustering method and deep learning with instantaneously trained neural networks (DITNN) strategy for lung cancer prediction using CT images obtained from an archive for cancer images. They implemented various preprocessing techniques, and the results showed that their method has an accuracy of 98.42% and a classification error of 0.038. Wang et al. (2020) developed a new residual neural network used to identify the pathological type of lung cancer using CT images data. The proposed model is trained on a public dataset using the transfer learning approach. The experiment yielded results of accuracy of 85.71%. Khamparia et al. (2020) designed an integrated Bhattacharya–GA-based model for feature selection and target gene identification. Their integrated Bhattacharya coefficient and genetic algorithm (GA) were applied to pick features and computed fitness using ensemble outputs from several classifiers, using deep learning approaches. Their proposed method outperformed other techniques in the prediction and diagnosis of neuromuscular disorders.

Besides, Zheng et al. (2021) proposed a combination of radiology analysis and malignancy evaluation network (R2MNet) to evaluate pulmonary nodule malignancy by radiology features analysis using the LIDC-IDRI dataset. Their results presented that the proposed method scored an AUC of 96.27% and 97.52% on nodule radiology analysis and nodule malignancy evaluation, respectively. Additionally, Bhandary et al. (2020) provided a deep Learning approach for investigating lung pneumonia and cancer. Two models proposed a modified AlexNet (MAN) and a fusion of hand-crafted and learned features in the MAN to improve classification accuracy during lung cancer assessment. Their results showed that the proposed model attained 97.27% accuracy.

Alagarsamy et al. (2021) used the crow search algorithm, an automated meta-heuristic-based optimization technique for lung cancer prediction. The results achieved a sensitivity of 99.12% compared to other methods. Furthermore, Sun et al. (2017) evaluated the effectiveness of standard computer-aided diagnosis (CADx) systems utilizing hand-crafted features to the use of deep learning structure for extracting features automatically for diagnosis of lung nodule CT image. They assessed the performance using a 10-fold cross-validation method. The results showed that CNN outperformed other models with an AUC of 0.899.

Oyelade & Ezugwu (2021c) proposed a new metaheuristic algorithm based on the propagation model of the deadly Ebola virus and its associated disease. They named their new algorithm Ebola optimization algorithm (EOSA). They evaluated the proposed model using two kind of benchmark functions: 47 classical and more than 44 constrained IEEE CEC-benchmark functions. Their results showed that EOSA can compete with the state-of-the-art methods such as artificial bee colony (ABC), GA, particle swarm algorithms (PSO) based on scalability, convergence, and sensitivity analyses.

Moreover, Miah & Yousuf (2015) proposed a method for lung cancer detection in two stages, including passing the image into preprocessing steps and segmenting and extracting the feature from the image samples. Their suggested method identified lung cancer in its early stages. Their proposed system consists of image acquisition, preprocessing, binarization, thresholding, segmentation, feature extraction, and neural network detection. Their results showed that the proposed system achieved an accuracy of 96.67%. Maleki et al. (2021) presented an approach based on a k-Nearest-Neighbors and GA algorithm for identifying the stage of lung cancer. The feature extraction is achieved using the GA algorithm. The proposed method scored an accuracy of 100% on the lung cancer database.

Bhatia et al. (2019) presented a technique for detecting lung cancer from CT data using deep residual learning. Preprocessing and feature extraction were carried out in order to make the data ready for further analysis. They used XGBoost and Random forest methods, and then the ensemble approach is implemented based on the individual classifiers. Their results outperformed previous studies and achieved 84% on LIDC-IRDI dataset. Song et al. (2017) developed CNN, DNN, and SAE deep neural networks for lung cancer classification using CT scan images. They assessed the methods using the LIDC-IDRI database. Their results showed that the CNN obtained the best performance with an accuracy, sensitivity, and specificity of 84.15%, 83.96%, and 84.32%, respectively compared to the other three networks.

Coudray et al. (2018) designed a CNN architecture based on inception V3 on whole-slide images from the cancer genome atlas to classify lung cancer into its subtypes. Their results showed that the model achieved an AUC of 0.97. Moreover, Chon et al. (2017) used a dataset from the Kaggle data science bowl 2017 to establish a CAD system for classifying lung cancer CT scans with unmarked nodules. They used a vanilla 3D CNN, and a Googlenet-based 3D CNN. The baseline classifier of their methods is a linear classifier. Their results showed that 3D Googlenet outperformed other methods with an AUC of 75.7%. Lu et al. (2021) proposed a new CNN architecture for the optimal detection of lung cancer. They used a metaheuristic method called marine predators for improved network accuracy and optimal design. Furthermore, the approach has been applied to the RIDER dataset. Their results have been compared with pre-trained deep networks, such as AlexNet, CNN ResNet-18, VGG-19, and GoogLeNet. Their proposed MPA-based approach scored an accuracy of 93.4%, sensitivity of 98.4%, and specificity of 97.1%.

Senthil Kumar et al. (2019) investigated and implemented five optimization algorithms, namely, inertia-weighted particle swarm optimization, k-median clustering, guaranteed convergence particle swarm optimization (GCPSO), k-means clustering, and particle swarm optimization, to extract the tumor from the lung cancer image. Also, they prepare the data for further analysis. The results showed that the GCPSO has the highest accuracy of 95.89%. Besides, Shan & Rezaei (2021) proposed an automatic and optimized computer-aided recognition for lung cancer. Feature selection was carried out based on improved thermal exchange optimization (ITEO). Their results revealed that the proposed method achieved 92.27% accuracy.

Abugabah et al. (2020) applied a meta-heuristic optimization algorithm for lung cancer images. The system efficiency was assessed using the F1 score value, which indi-

cates that the system ensures high accuracy of 98.9% in ELT-COPD and 98.9% in NIH clinical datasets. Priyadharshini et al. (2021) introduced a bat-inspired metaheuristic based on CNN algorithms for CAD-based Lung Cancer detection. The discrete wavelet transform (DWT) was used to decompose the image. The suggested method has been validated using publicly available LIDC-IDRI data. The result showed that the proposed method achieved an accuracy of 97.43% with a small classification error rate of 2.57% in lung cancer prediction. Li et al. (2019) used metaheuristic techniques to optimize the rebalancing of the imbalanced class distributed in order to apply it in the feature selection method for dimension reduction in clinical X-ray image datasets. Their result showed that the Bat algorithm's proposed technique achieved 94.6% classification accuracy with 88.3% Kappa using the lung X-ray dataset. Moreover, Al-Huseiny & Sajit (2021) proposed DNN model to classify lung cancer images with malignant nodules. They implement transfer learning to readjust GoogleNet DNN to learn from the images. Their results showed that an accuracy of 94.38% was achieved by the proposed model. A summary of the previous related studies is presented in Table 2.1 below.

Table 2.1: A comparative summary of related existing studies.

Author	Preprocessing	Methods	Findings
Revathi et al. (2020)	Noise removal Gabor filter	Multi-swarm particle swarm optimizer	Accuracy: 98%
Das et al. (2020)	-	Velocity-Enhanced Whale Optimization Algorithm	Accuracy: 84%
Shakeel et al. (2019)	Noise removal histogram of images quality enhance images	deep learning with instantaneously trained neural networks (DITNN)	Accuracy: 98.42%
Wang et al. (2020)	Removes unreadable scans labelling tumor areas	a novel residual neural network	Accuracy: 85.71%
Zheng et al. (2021)	Normalization Extraction Resample Crop	combination of radiology analysis and malignancy evaluation network (R2MNet)	Accuracy: 89.90% AUC: 96.27%
Khamparia et al. (2020)	-	deep network ensemble method	Accuracy: 87.35%

Continued on next page

Table 2.1 – continued from previous page

Author	Preprocessing	Methods	Findings
Bhandary et al. (2020)	Image separation Nodule segmentation	Deep Learning approach	Accuracy: 97.27%
Alagarsamy et al. (2021)	-	metaheuristic based optimization	Sensitivity: 99.12%
Sun et al. (2017)	Segmentation	CNN deep belief network (DBN) stacked denoising autoencoder (SDAE)	AUC: 89.9%
Oyelade & Ezugwu (2021c)	-	New metaheuristic algorithm (EOSA)	Friedman test for EOSA is 1.60 which is ranked as the first compared to others
Miah & Yousuf (2015)	Gray Scale Conversion Normalization Noise Reduction Binary Image Remove Unwanted Portion of the Image	Neural network	Accuracy: 96.67%
Maleki et al. (2021)	Remove missing values fill the missing values clean the dataset	Approach based on k Nearest Neighbors and GA algorithm	Accuracy: 100%
Bhatia et al. (2019)	Applications of region growing and morphological operations	ResNet XGBoost	Accuracy: 84%
Song et al. (2017)	Data Augmentation	CNN DNN SAE deep neural networks	Accuracy: 84.15% Sensitivity: 83.96% Specificity: 84.32%

Continued on next page

Table 2.1 – continued from previous page

Author	Preprocessing	Methods	Findings
Coudray et al. (2018)	-	CNN architecture based on inception V3	AUC: 97%
Chon et al. (2017)	convert the pixel values thresholding clustering (K-means and Meanshift) Watershed normalize spline interpolation zero-center the data	vanilla 3D CNN Googlenet based 3D CNN	AUC: 75.7%
Lu et al. (2021)	Noise Removal Image Level Balancing	metaheuristic method called marine predators AlexNet CNN ResNet-18 VGG-19 GoogLeNet	Accuracy: 93.4% Sensitivity: 98.4% Specificity: 97.1%
Senthil Kumar et al. (2019)	Median, Average, Adaptive median, Adaptive histogram equalization	inertia-weighted particle swarm optimization k-median clustering guaranteed convergence particle swarm optimization (GCPSO) k-means clustering and particle swarm optimization	Accuracy: 95.89%
Shan & Rezaei (2021)	normalizing and denoising the input images	Improved Thermal Exchange Optimization	Accuracy: 92.27%
Abugabah et al. (2020)	min–max normalization	harmony optimized modular neural network	Accuracy: 98.9%
Priyadharshini et al. (2021)	Segmentation	bat-inspired metaheuristic based on CNN algorithms	Accuracy: 97.43%

Continued on next page

Table 2.1 – continued from previous page

Author	Preprocessing	Methods	Findings
Li et al. (2019)	Segmentation	Bat algorithm	Accuracy: 94.6% Kappa: 88.3%
Al-Huseiny & Sajit (2021)	Gabor filter Morphological operation normalised	DNN model with transfer learning	Accuracy: 94.38%

2.5 Research Gap

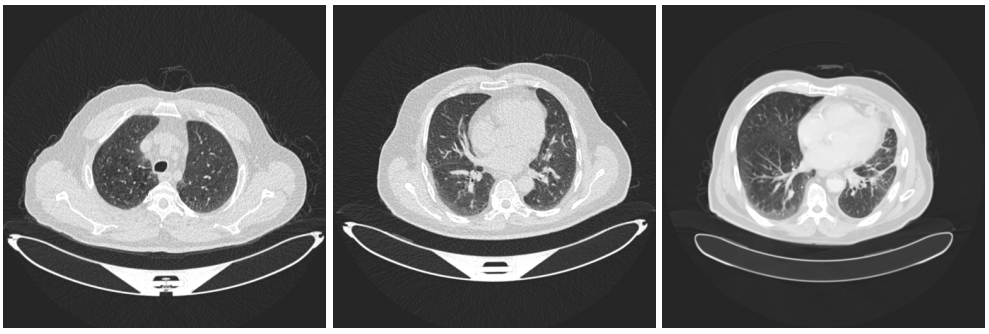
In the cause of reviewing several related literature for this thesis, many gaps were identified concerning using Deep Learning models for the task at hand. The training process of DNN models has particular inefficiency due to the very long training time demanded, and there are potentially many parameters in the DNN model structure and the high dimensionality of the feature in the dataset used for training the DNN model. To address these, there is need to optimize the CNN model using approximate optimization method. Several metaheuristic algorithms have been proposed, implemented and applied to address these problems. However, the aforementioned critical challenges of the efficient and effective classification of lung cancer using deep learning models remains unresolved. This study therefore aims to improved the performance of DL models by using a novel optimization algorithm based on the biological process of Ebola disease.

Chapter 3

Methodology

3.1 Dataset

The Iraq-Oncology Teaching Hospital/ National Center for Cancer Diseases (IQ-OTH/NCCD) lung cancer dataset Kerneler was used for experimenting the method proposed in this study. This dataset of CT images was collected from two specialist hospitals for three months in 2019. The data is composed of CT scans taken from lung cancer patients diagnosed in various stages and those reporting with normal cases. The data consist of 1097 samples (images) taken from one hundred and ten (110) cases categorized into three different classes: normal, benign, and malignant. One hundred and twenty (120) samples are benign, five hundred and sixty-one (561) samples are malignant, and four hundred and sixteen (416) are normal samples. Figure 3.1 shows random samples of the original dataset.



(a) Normal. **(b)** Benign. **(c)** Malignant.

Figure 3.1: Samples of images with (a) normal, (b) benign, and (c) malignant labels from original dataset.

3.2 Data Preparation

The preparation of the data, also known as preprocessing, describes any processing that makes and prepares the raw data to be ready for another task such as classification, prediction, and clustering. To enhance performance of deep learning models, different preprocessing techniques have been proposed. In this study, the preprocessing phase includes many functions for manipulating the images into a suitable form for further analysis. Firstly, we downloaded the data from Kaggle and then read it using python. Then we applied the following image preprocessing methods: image resizing, conversion of image to grayscale mode, gaussian blur filter, segmentation, normalization, erosion, noise removal, and wavelet transform into the lung cancer images. There are many studies that have used preprocessing techniques similar to the approach implemented in this thesis such as Rajakumari & Kalaivani (2021); Miah & Yousuf (2015). Figure 3.2 shows the procedure for the preprocessing methods implemented in our study.

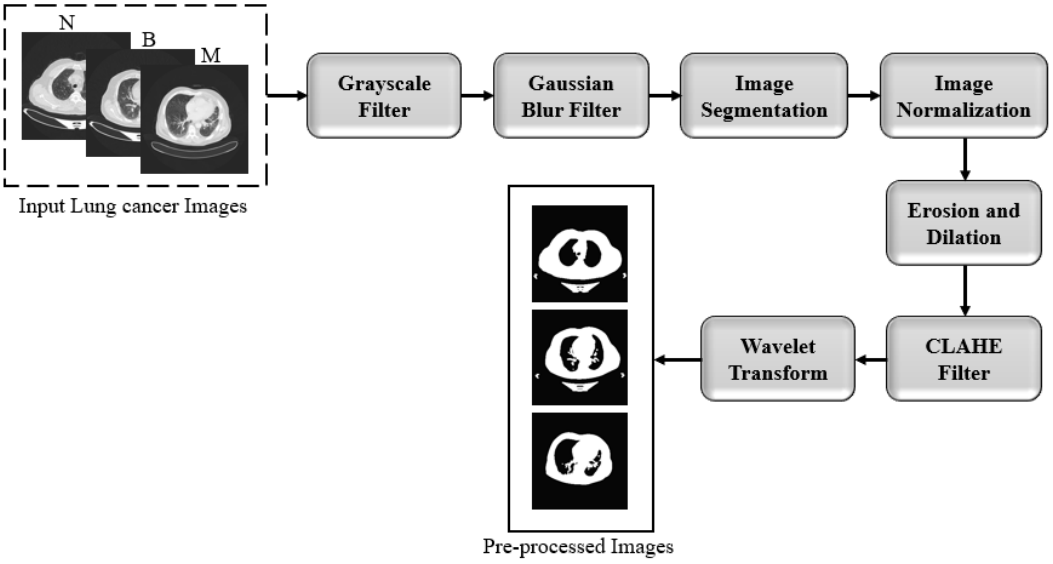


Figure 3.2: The data preprocessing procedure.

Thereafter, we partitioned the preprocessed data into 80% as training set and 20% as testing set. Then, we designed the proposed CNN architecture to compute the solution vector which is further optimized by the metaheuristic algorithms as shown in Figure 3.3 below.

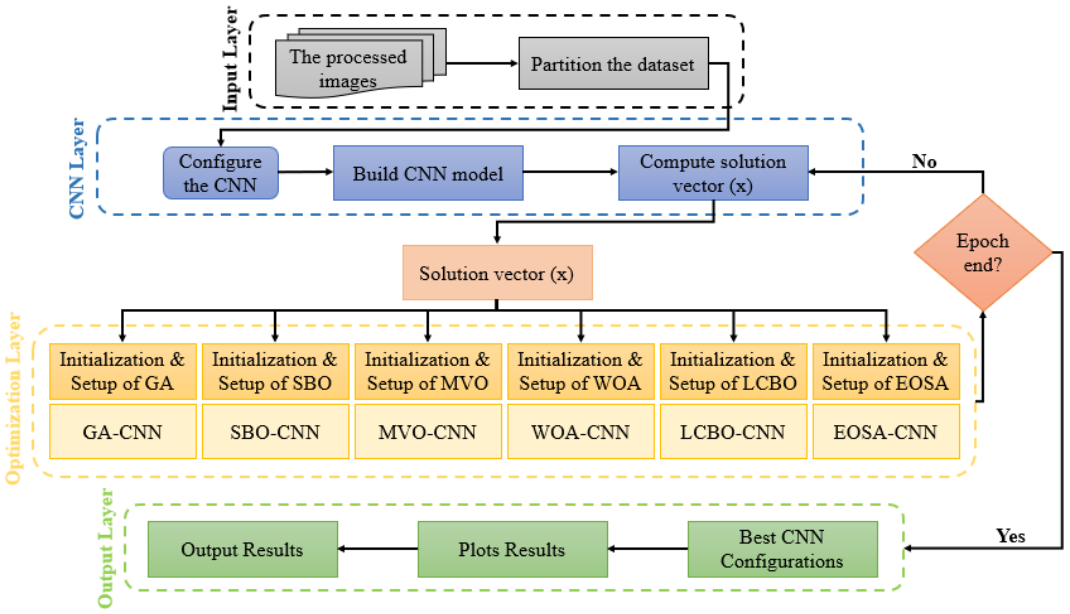


Figure 3.3: The proposed methodology.

3.2.1 Transforming the Lung Cancer Images into Grayscale Mode

Grayscale image contains only a single channel, and pixels represent the intensity information of the light, e.g., the images only include the black (0), white (255), and gray colors, where gray has multiple levels. Usually, the grayscale intensity uses 8 bits. These bits are a combination of eight binary digits denotes 256 various shades of gray from black to white. The image processing helps convert the image into grayscale using a threshold value. The thresholding is a method for adjusting the number of gray degree values in images. The pixels of the colored image has a matrix value, which is consists of red, green, and blue (r, g, b) Irawanto et al. (2022). The colored image can be converted into grayscale using the following steps.

- To calculate the grayscale value, we use $S = \frac{r*g*b}{3}$, where S is the grayscale value.
- To calculate the thresholding value, we use $x = \frac{w}{b}$, where x indicates the gray degree value after thresholding, w indicates the gray degree value before thresholding, and b represents the intended quantization values.

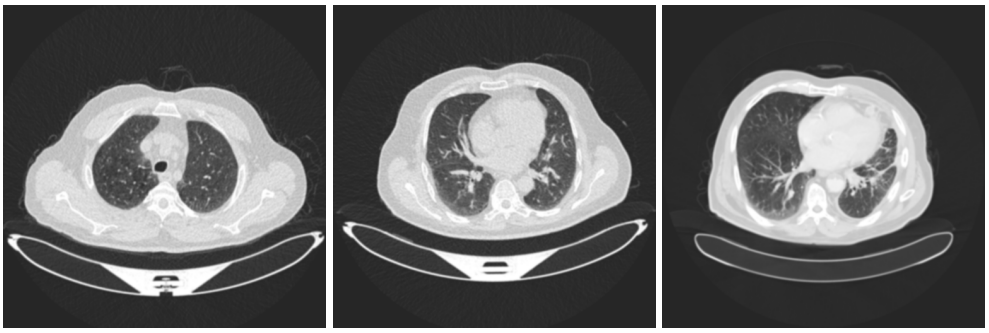
We used *cvtColor()* function in the OpenCV library to convert the lung cancer images into grayscale. The grayscale images are shown in Figure 3.4 below.

(a) Normal. **(b)** Benign. **(c)** Malignant.

Figure 3.4: Illustrate the transformed images of normal, benign, and malignant samples into grayscale.

3.2.2 Gaussian Blur Filter

The Gaussian blur is a linear filter type technique that helps image processing by implementing smoothing and blurring effects to remove the noise. It calculates the weighted average of pixel intensities at adjoining positions. These weights reduce concerning spatial distance to the center. This filter produces a blurred image with a minimal presence of noise Ibrahim et al. (2021); Rajasenbagam & Jeyanthi (2020); Chaturvedi et al. (2021). We have utilized the *GaussianBlur()* function of the OpenCV library in python. Figure 3.5 below describes the results of the Gaussian blur filter on normal, benign, and malignant lung images.



(a) Normal. **(b)** Benign. **(c)** Malignant.

Figure 3.5: Effect of applying Gaussian blur filter on the lung cancer samples with normal, benign, and malignant labels.

3.2.3 Otsu's Thresholding

Segmentation is considered as a fundamental process in preparing medical images computer-aided application Shoaib et al. (2013). Otsu's thresholding technique uses a threshold value that divides the image into foreground and background. The threshold value increases gradually to reach the maximum variance between the

pixels of two classes. Thresholding is an easy segmentation method that extracts the area of interest. This technique helps split a digital image into several separate parts so that pixels in each location have comparable visual qualities is known as segmentation. The idea is to simplify or modify an image's representation to analyze it further Kornilov & Safonov (2018). A threshold is a technique of separating foreground pixels from background pixels. The processed image is subjected to a gray threshold using Ostu method, and the image's intensity is altered halfway between the lowest and highest of the native intensity Rajakumari & Kalaivani (2021); Abdullah et al. (2019). Therefore, otsu's threshold technique segments the entity from the image. Otsu's method's main objective is to give the optimum threshold value. The computation of the threshold value is done by clustering the pixels into two categories, C1 and C2, with bimodal histograms. Also, it reduces the intraclass variance by picking an appropriate threshold value. We have used *threshold* function in python for the implementation. Figure 3.6 demonstrates the effects of the Otsu's method on the lung cancer images.

(a) Normal. **(b)** Benign. **(c)** Malignant.

Figure 3.6: An Illustration on the outcome of the Otsu's method on samples with normal, benign, and malignant labels.

3.2.4 Image Normalization

Image normalization modifies the range of pixel intensity values. In other words, normalization ensures that the pixel intensity values range is within range. This process provides optimal comparison across the lung cancer images. The data is normalized by subtracting the mean from each pixel and dividing the result by the standard deviation Depeursinge et al. (2017). We used *normalize* function in python for normalizing the lung cancer images as can be seen in Figure 3.7).

(a) Normal. **(b)** Benign. **(c)** Malignant.

Figure 3.7: An illustration of applying normalization to the lung cancer samples with normal, benign, and malignant labels.

3.2.5 Image Erosion and Dilation

Erosion and dilation are the essential morphological procedures in image processing. This process aims to extract the most relevant structure of the image viewed as a set through its subgraph representation Siddiqi et al. (2009); Tambe et al. (2013). Erosion is the procedure of eradicating edge points and shrinking the edge to the inside. It eliminates small and senseless entities. Erosion utilizes the structural components and the ”and” operation to examine every pixel of the image Zhang et al. (2021a); Soille (2013). The mathematical equation of the erosion process is defined as follows:

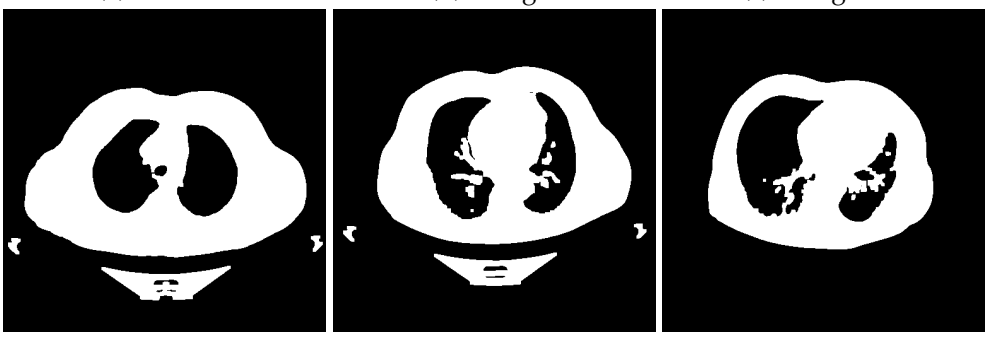
$$Y = A \ominus B = \{x, y | (B)_{xy} \subseteq A\} \tag{3.1}$$

where, Y is a binary image, B is a template operator, and A is the original image to be processed. While dilation is the process of combining all the background points in contact with the object into the object to expand the boundary outward, it fills gaps in entities Zhang et al. (2021a); Siddiqi et al. (2009). The dilation can be defined mathematically by:

$$Y = A \oplus B = \{y : B(y) \cap E \neq \Phi\} \tag{3.2}$$

where, Y is a binary image, B is a template operator, and E is the original image to be processed. The processed lung cancer images after the erosion and dilation can be viewed in Figure 3.8 below

(a) Normal. **(b)** Benign. **(c)** Malignant.



(d) Normal. **(e)** Benign. **(f)** Malignant.

Figure 3.8: An illustration on the outcome of erosion method in (a), (b), and (c) images. Images in (d), (e), and (f) are the outcome of dilation method on images for normal, benign, and malignant labels.

3.2.6 Noise Removal using CLAHE Filter

The presence of noise in an image is represented by random variation of color information or brightness in images. This noise may come from various sources, which erode image quality Kaur (2015). In this study, contrast-limited adaptive histogram equalization (CLAHE) filter to remove the unwanted noise. CLAHE is a variant of Adaptive histogram equalization (AHE), which takes care of over-amplification of the contrast. Moreover, CLAHE is an image processing technique that increases the contrast in local images by setting a maximum threshold value. It splits the local image contrast in a symmetrical grid known as the area size consisting of corner region (CR), border region (BR), and inner region (IR). There are several characteristic values for each value of the neighborhood. This is computed by obtaining a new gray level on each local grid using the cumulative function distribution of histograms used in each local region (i, j) Irawanto et al. (2022). The distribution can be calculated using

$$K = \frac{N-1}{M}. \qquad \sum_{k=0}^{n} h(K); n = 1, 2, \ldots, N-1. \tag{3.3}$$

where M is the number of pixels, N is greyscale value, and $h(k)$ is histogram on grey value. Furthermore, we applied CLAHE method to find the maximal threshold limit on the image using the equation below:

$$\beta = \frac{M}{N}(1 + \frac{\alpha}{100}(S_{max} - 1)) \tag{3.4}$$

where M is width of region size, N is greyscale value, and α represents the minimum and maximum limit values of histograms in [0, 100]. We used *CLAHE* function in python to remove the noise. The result of CLAHE filter can be seen in Figure 3.9 below.

(a) Normal. **(b)** Benign. **(c)** Malignant.

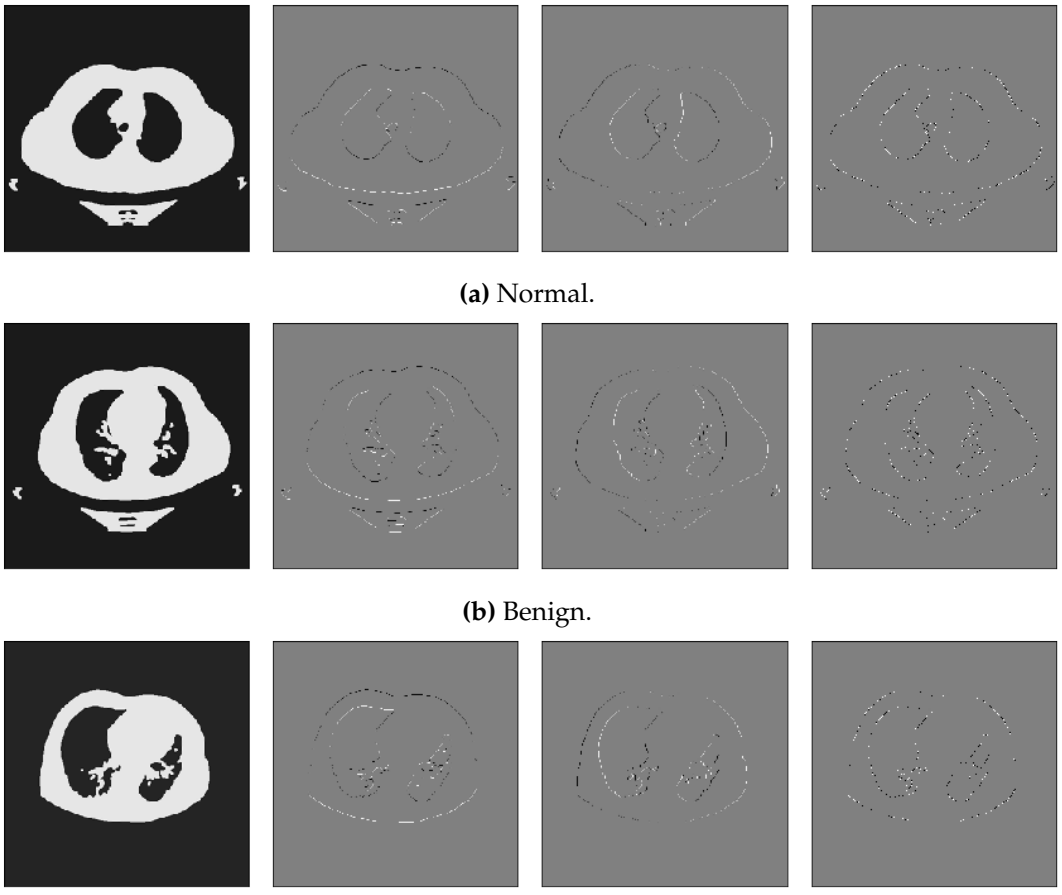
Figure 3.9: A demonstration on the outcome of applying CLAHE filter on image samples with normal, benign, and malignant labels.

3.2.7 Wavelet Transform

The wavelet analysis is a multivariate analysis that is used in preprocessing of medical images Hassen & Zakour (2019); Boix & Cantó (2010). This transform analysis provides a different representation by decomposing and compressing the images. The wavelet has two decomposition levels; the first level produces two coefficient vectors, namely approximation and detail coefficient. The latter represents low and high-frequency contents in the image. In the second level when wavelet is applied to the image, the approximation coefficient produces another approximation and detail coefficient but with lengths equal to half of the initial approximation in the first level.

The advantage of using a wavelet is that it uses multi-resolution approach instead of scanning whole signals via the same window. Various parts of the wave are viewed via a different sized window, where high-frequency components of the signal use a small window to give reasonable time resolution. In contrast, the low-frequency parts use a large window to extract good frequency information Prasad et al. (2016); Sridhar et al. (2014); Khatami et al. (2017); Rajakumari & Kalaivani (2021).

In this study, we used the bi-orthogonal family variant of wavelet which is the *pywt.dwt2* function in python. The wavelet output is depicted in Figure 3.10 below and is decomposed into four quadrants with different interpretations namely: LL (low-low), HL (high-low), LH (low-high) and HH (high-high). LL is defined as the approximated version of the original at half the resolution, while HH is identified as the area where the edges of the original image are represented diagonally. Furthermore, HL is the image's upper right, representing the horizontal edges. LH is the lower-left that consists of most of the vertical edges.



(a) Normal.

(b) Benign.

(c) Malignant.

Figure 3.10: A representation of decomposition effect of wavelet filter function on image samples with normal, benign, and malignant labels.

Here we selected the LL part for further analysis as shown in Figure 3.11. The LL indicates the upper-left quadrant that has all coefficients, which were filtered by the analysis low pass filter $h\tilde{}$ along the rows and the corresponding columns.

(a) Normal. **(b)** Benign. **(c)** Malignant.

Figure 3.11: The output for the LL compartment from the effect of applying the wavelet filter function to the image samples with normal, benign, and malignant labels.

Several techniques have been developed to diagnose and detect lung cancer using CT images data. These techniques have many advantages and disadvantages. This study proposed an optimized CNN using a metaheuristic algorithm called EOSA, which is based on the propagation model of the deadly Ebola virus and its related disease. We compare its performance with five metaheuristic algorithms and the traditional CNN using IQ-OTH/NCCD dataset. In the next section, we present the background and details of each algorithm.

3.3 Convolutional Neural Network (CNN)

The CNN is a deep learning algorithm that contains several layers in between the input and output layers. It has been applied to image analysis and classification Mohammed et al. (2021); Elbashir et al. (2019); Bengio (2009). Moreover, CNN is a mathematical model designed from three layers: convolution, pooling, and fully connected layers. The CNN conduct feature extraction using the convolution and pooling layers, while the the fully connected layers map the extracted features into the final output Yamashita et al. (2018).

CNN needs much lower pre-processing by learning the filters that capture temporal and special dependencies in an image instead of hand-crafting features extraction procedure. It has been implemented successfully in various domains such as bioinformatics, computer vision, drug design, and medical image analysis. CNN usually outperformed expert personals' performance Oyelade & Ezugwu (2021b); Ciregan et al. (2012). In this study we proposed a CNN architecture in Figure 3.12 for the classification task on lung cancer images, while the algorithmic design is presented in Algorithm listing 1.

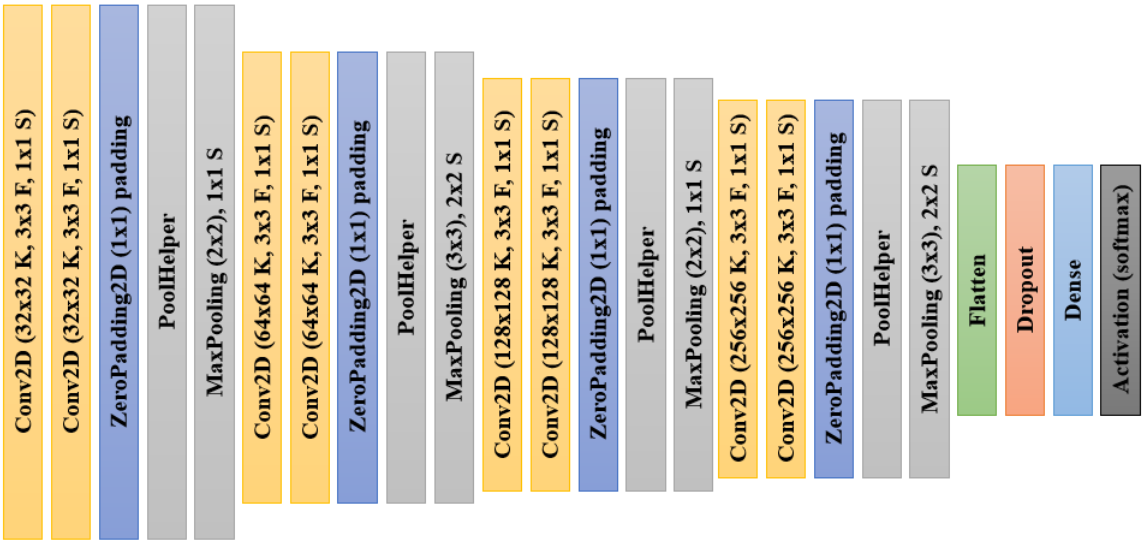


Figure 3.12: The architecture of the proposed CNN model for lung cancer detection, where the notations F, K, and S, indicates the filters, kernels, and strides respectively in the architecture.

Algorithm 1: CNN Architecture Desing

Result: model of CNN graph

numclasses, numblocks, kSize cnnmodel **Data:** Dataset *mias*

```
cnnmodel = ∅ ;                                  // initialize the model
blk=0;
n=5;
while blk ≤ numblocks do
    kcount=2^n;
    cnnmodel ← layer2D(kSize, kcount, relu);
    cnnmodel ← zeropad(1);
    if blk %= 2 then
        cnnModel ← maxpool(2);
    end
    else
        cnnmodel ← maxpool(3);
    end
    n+=1;
end
cnnmodel ← avgpool(2);
cnnmodel ← flatten();
cnnmodel ← dropout(0.5);
cnnmodel ← dense(softmax, numclasses);
```

3.4 The Metaheuristic Optimized CNN Algorithms

A metaheuristic is a well-known search strategy that is designed to find an optimal solution for a complex and challenging optimization problem, particularly with incomplete or inadequate data with limited computation capacity Balan; Rere et al. (2016); Bandaru & Deb (2016). Recently, metaheuristic has emerged for use in solving various optimization problems Yang (2011). Many optimization problems are difficult to formulate, as a result, presenting a challenge to applying them to find optimal solutions in artificial intelligence and machine learning. However, it still continue to play a substantial role in discovering optimal and near-optimal solutions that are difficult to solve using conventional optimization methods.

In this study, we used metaheuristic to explore and improve the performance of the CNN model in a reasonable computational time. To conduct this task, we obtain the solution vector of the CNN model and update it in every iteration when

training the metaheuristic algorithm. The updated solution vector is calculated for several epochs, leading to updated weights and biases for the CNN layers. Moreover, the CNN model calculates the weights and biases that can compute the loss function used in this work. In this work, we used different categories of metaheuristic algorithms, namely, GA (evolutionary-based), WOA (swarm-based), MVO (physics-based), SBO (biology-based), LCBO (human-based), and EOSA (biology-based). These algorithms are described in the following sections below.

3.4.1 Design of the Genetic Algorithm CNN (GA-CNN)

The GA was first presented by John Holland in the early 1970s Park & Monahan (2019); Li & Weinberg (2003). It uses natural selection, which is the process that drives biological evolution for solving optimization problems Oyelade & Ezugwu (2021b). In addition, the GA is known as an evolutionary algorithm that uses the concept of survival of the fittest. GA is a randomized heuristic search and optimization algorithm implemented in various applications. This algorithm begins with a group of solutions known as population. These solutions (offspring) are used to construct a new population that is expected to be better compared to the old one Chadha & Singh (2012). The best solution is selected based on its fitness and suitability. This process is iterated until some condition is satisfied. We used the GA because it produces and generates a population of solutions, which contains the best solution close to the optimal one in each epoch. The steps below describe the mechanism of the GA, while the algorithmic design is presented in Algorithm listing 2.

1. We initialized the parameters namely population size, population percentage mutated, mutation rate, penalty rate, the maximum generation, and the number of generations.
2. We generated an initial population of the chromosomes using a random model.
3. The reliability of the population is calculated.
4. The fitness of all chromosomes in the population is calculated.
5. We initialized an empty successor population.
6. Repeat the following steps (a-d) below until chromosomes have been created.
   (a) We used proportional fitness selection to choose two chromosomes from the source population.
   (b) Create a child chromosome using the one-point crossover to the two chromosomes with a crossover rate.

(c) Use consistent mutation for the child with a given mutation rate.

(d) Add the child to the previously initialized successor population.

7. creating the successor population takes over from the source population.

8. Terminate if the number of iteration is achieved or return to Step 2.

Algorithm 2: Algorithm of the GA-CNN

Result: optimized CNN model

```
objfunc, lb, ub, batch, epoch, pc, pm solution, bestfit, losstrain initialize IdFit,
IdMinProb ;
model=buildcnn();
weights=model.weights();
popsize=len(reshape(weights));
pop=createsolution(popsize);
gbest = gbestsolution(pop, IdFit, IdMinProb);
while epoch not exhausted do
    nextpop = [];
    while e in epoch do
        fitlist = [];
        for item in pop do
            fitlist ← item[IdFit];
        end
        c1 = roulettewheelselection(fitlist);
        c2 = roulettewheelselection(fitlist);
        w1 = pop[c1][IdPos];
        w2 = pop[c2][IdPos];
        if uniform() < pc then
            w1, w2 = crossoverarthmeticrecombination(w1, w2)
        end
        if uniform(0, 1, propsize) < pm then
            w1 = uniform(lb, ub, popsize)
        end
        if uniform() < pc then
            w2 = uniform(lb, ub, popsize)
        end
        nextpop ← w1;
        nextpop ← w2;
    end
    gbest = updategbest(nextpop,IdMinProb, gbest);
    losstrain ← gbest[IdFit];
end
solution = gbest;
return gbest[IdPos], gbest[IdFit], losstrain;
```

In algorithm 2, the weights size is used to initialize the population size and generate

the possible solutions. The iteration continues until the best solutions is obtained. Finally, the algorithm returns the global fitting best and the values of the loss functions.

3.4.2 Design of the Whale Optimization Algorithm CNN (WOA-CNN)

The WOA was presented by Mirjalili and Lewis in 2016 Mirjalili & Lewis (2016). It is a new approach for solving optimization problems that simulate the hunting behavior with random or best search agents to track the target and uses a spiral to mimic the bubble-net attacking mechanism of humpback whales Mirjalili & Lewis (2016). The concept of the whale is that when it finds its victim, it creates a bubble net along the spiral way and moves upstream to the victim. This behavior concept is summarized in three steps: surrounding prey, bubble-net attack, and hunting prey Ning & Cao (2021); Jianhao et al. (2021). WOA has been confirmed to solve NP-Hard optimization problems such as CNN training. This study examines the performance of WOA in optimizing the training of the proposed CNN architecture. The following steps explains the WOA procedures, while the algorithmic design is presented in Algorithm listing 3.

1. Initialize the WOA population randomly.
2. Calculate the fitness function to evaluate each search agent.
3. Choose the optimal search agent and make it the current one.
4. While $t <$ number of iteration
   (a) for each solution
      i. Update a, A, C, and L.
      ii. update the position of the search agent according to A and p.

$$\begin{cases} If \quad p \geq 0.5 & \text{update the position of the current agent.} \\ If \quad p < 0.5 & \begin{cases} If \quad A \geq 1 & \text{update the position of the current agent by choosing the best search agent.} \\ If \quad A < 1 & \text{update the current search agent.} \end{cases} \end{cases}$$

   (b) Adjust any search agent that goes beyond the search space.
   (c) Calculate the fitness of each search agent.
   (d) In case of finding a better agent, update the current search agent.
5. Return best search agent.

Algorithm 3: Algorithm of the WOA-CNN

```
Result: optimized CNN model
objfunc, lb, ub, batch, epoch solution, bestfit, losstrain initialize IdFit,
IdMinProb ;
model=buildcnn();
weights=model.weights();
popsize=len(reshape(weights));
pop=createsolution(popsize);
gbest = gbestsolution(pop, IdFit, IdMinProb);
while e in epoch do
    a = (2 - 2 * e) / (epoch - 1);
    for i in popsize do
        A = 2 * a * rand - a;
        C = 2 * rand;
        l = uniform(-1, 1);
        p = 0.5;
        b = 1;
        if uniform() < p then
            if abs(A) < 1 then
                D = abs(C * gbest[IdPos] - pop[i][IdPos]);
                npos = gbest[IdPos] - A × D;
            else
            end
            xrand = newsolution();
            D = abs(C * rand[IdPos] - pop[i][IdPos]);
            npos = xrand[IdPos] - A × D;
        else
            D = abs(gbest[IdPos] - pop[i][IdPos]);
            npos = D × exp(b × l) × cos(2 × pi × l) + gbest[IdPos];
        end
        fit = fitnesspos(npos) pop[i] = npos[fit] if i % batch then
            gbest = updategbest(nextpop,IdMinProb, gbest);
        end
    end
    losstrain ← gbest[IdFit];
end
solution = gbest;
return gbest[IdPos], gbest[IdFit], losstrain;
```

The algorithm 3 is used to optimize the CNN model presented 1. The weight of the CNN model was used to initialize the population. The algorithm assumes that the initialized population is the global best solution. This algorithm searches the global optimum solution using encircling prey, searching for prey, and attacking the prey. In addition, the best solution is returned as a group of optimized weights for the CNN model. The best set of the loss values can be obtained through the parameter α which is linearly decreased from 2 to 0. Moreover, when $p < 0.5$ and $|A| < 1$ satisfied the exploitation stage is done, while the exploration stage is completed when $p \geq 0.5$.

3.4.3 Design of the Multiverse Optimizer CNN (MVO-CNN)

The big bang theory considered that the universe that we live in started with a tremendous explosion Khoury et al. (2002). This theory believed that there was nothing before the big bang. Moreover, recently, a well-known approach called multiverse states that assume there is more than one big bang, each of which yields a universe's birth Tegmark (2003). The MVO is inspired by this theory and it uses three main approaches namely, white hole, black hole, and wormhole Rosales-Muñoz et al. (2021); Mirjalili et al. (2016). Although the white hole has never been noticed in the universe that we live in, physicists believe that the Big Bang can be considered a primary element for the universe. In contrast, black holes have been seen frequently in the universe. The black hole attracts everything that comes through, including light beams, because of its extremely high gravitational force Davies (1978). Wormholes are defined as the tunnels that connect the various parts of the universe. These tunnels allow objects to travel immediately between any corners inside the universe or from one universe to another Benmessahel et al. (2020).

The MVO is a population-based algorithm that uses exploration and exploitation search strategies. In addition, MVO utilizes the white and black holes to explore the search spaces, and it uses the wormholes to exploit the search spaces Oyelade & Ezugwu (2021b). This algorithm deems all solutions as analogous to a universe, while each solution is an object in that universe Almomani (2021).
The following steps represent the MVO process, while the algorithmic design is presented in Algorithm listing 4.

1. Initialize the universes randomly.
2. Initialize the number of parameters.
3. While iteration not exhausted.

(a) Two parameters should be calculated (wormhole existence probability (WEP) and traveling distance rate (TDR)) as follows:

$$WEP = min + l \times \left(\frac{max - min}{L} \right)$$

where min is the minimum, max is the maximum, l denotes the current iteration, and L indicates the maximum iteration.

$$TDR = 1 - \frac{l^{1/p}}{L^{1/p}}$$

where p denotes the exploitation accuracy over iteration. The higher p, the sooner and more accurate exploitation search.

(b) for each object starting from index 1.

i. Assume r is a random variable followed the uniform distribution [0,1].

A. Update the position of solutions using the following equation:

$$x_i^j = \begin{cases} If \quad r < WEP & \begin{cases} If \quad r < 0.5 & X_j + TDR \times ((ub_j - lb_j) \times r + lb_j) \\ Else & X_j - TDR \times ((ub_j - lb_j) \times r + lb_j) \end{cases} \\ Else & x_i^j \end{cases}$$

where, X_j denotes j^{th} parameter of the best universe, lb_j is the lower bound of j^{th} variable, ub_j is the upper bound of the j^{th} variable, x_i^j indicates the j^{th} parameter of the i^{th} universe.

ii. Obtain the fittest position using the black hole position.

iii. Calculate the position updates in the optimal universe.

(c) Store the global best.

4. Return all global best.

Algorithm 4: Algorithm of the MVO-CNN

```
Result: optimized CNN model
objfunc, lb, ub, batch, epoch, wminmax solution, bestfit, losstrain initialize IdFit,
IdMinProb ;
model=buildcnn();
weights=model.weights();
popsize=len(reshape(weights));
pop=createsolution(popsize);
gbest = gbestsolution(pop, IdFit, IdMinProb);
while e in epoch do
    wep = wminmax[1] - (e + 1) × ((wminmax[1] - wminmax[0]) / epoch);
    tdr = 1 - (e + 1) ** (1.0 / 6) / epoch ** (1.0 / 6);
    for i in popsize do
        if uniform() < wep then
            if uniform() < 0.5 then
                fitlist=[] for item in pop do
                    fitlist ← item[IdFit];
                end
                whiteholeid =roulettewheelselection(fitlist);
                bholepos1 = pop[i][IdPos] + tdr × normal(0, 1) ×
                 (pop[whiteholeid][IDPos] - pop[i][IdPos]);
                bholepos2 = gbest[IdPos] + tdr × normal(0, 1) × (gbest[IdPos] -
                 pop[i][IdPos]);
                if uniform(0, 1, propsize) < 0.5 then
                    bholepos = bholepos1;
                end
                else
                    bholepos = bholepos2;
                end
            else
                bholepos = levyflight(e + 1, pop[i][IdPos], gbest[IdPos]);
            end
        else
            bholepos = uniform(lb, ub);
        end
        fit = fitnesspos(bholepos);
        if fit ¡ pop[i][IdFit] then
            pop[i] = [bholepos, fit];
        end
        if i % batch then
            gbest = updategbest(nextpop,IdMinProb, gbest);
        end
    end
    losstrain ← gbest[IdFit];
end
solution = gbest;
return gbest[IdPos], gbest[IdFit], losstrain;
```

The algorithm 4 considers a search space using the vector solution from the CNN model. Before the white- and black-holes positions are searched for within the weights of the CNN, the value of the wormhole existence probability and traveling distance rate are calculated. The global best solution and loss values are generated before the loop terminates.

3.4.4 Design of the Satin Bowerbird Optimization CNN (SBO-CNN)

The SBO is a new metaheuristic algorithm that works based on the concept and the mechanism of the stain birds. This pick the female birds for mating in specialized stick structures, called bowers Zhang et al. (2021b); Hemeida et al. (2021); Moosavi & Bardsiri (2017). The bowers are adorned with flowers, feathers, and berries which is essential for female birds to determine on the male bird for mating Ali et al. (2019). The males compete by imitating the females using the gorgeous decorated bower and dancing with loud vocalization and presentation surrounding the bowers. The females visit many bowers before determining their partner for breeding, and they desire the males that exhibit high passion. In addition, not all males succeed in building and maintaining the bowers. Therefore, this leads to substantial variation in successful mating. The SBO has been applied in various fields due to its flexibility and efficiency in features selection; thus, it is capable of relatively competitive accuracy Oyelade & Ezugwu (2021b). The SBO algorithm was constructed based on the following steps on the principle of the stain birds' lifestyle, while the algorithmic design is presented in Algorithm listing 5.

1. Randomly generate a population of bowers.
2. Compute the cost of each bowers.
3. Locate the best bowers.
4. While the ending criterion is not satisfied.
   (a) Compute the probability of each bower using the following equation:

$$Prob_i = \frac{fit_i}{\sum_{n=1}^{N} fit_n}$$

$$fit_i = \begin{cases} if \quad f(x_i) \geq 0 & \frac{1}{1+f(x_i)} \\ Else & 1 + |f(x_i)| \end{cases}$$

where N indicates the population size of the bowers, fit_i denotes the fitness value of the i^{th} solution, and $f(x_i)$ indicates the fitness value of the

i^{th} bower.

(b) For each bower

i. For each element of bower

- Select a bower using the roulette wheel.
- Calcuate λ_k using

$$\lambda_k = \frac{\alpha}{1 + p_j}$$

where α is a constant that denotes the greatest step size, p_j indicates the probability that calculated in step (a) above, $P_j \in (0, 1)$.

- Compute and update the bower position using

$$b_{ik}^{new} = b_{ik}^{old} + \lambda_k \left(\left(\frac{b_{ik} + b_{elite,k}}{2} \right) - b_{ik}^{old} \right)$$

where b_i is the i^{th} bower (solution vector), b_j indicates the target solution among all solutions in the current iteration, j is computed by the roulette wheel, b_{ik} denotes the k^{th} member of dimensions, b_{elite} is the best fitness value in the current iteration.

5. Calculate the cost of all bowers.

6. Update the elite of bower.

7. Return the best bower.

Algorithm 5: Algorithm of the SBO-CNN

```
Result: optimized CNN model
objfunc, lb, ub, batch, epoch, alpha, pm, sigma, lamda solution, bestfit, losstrain
initialize IdFit, IdMinProb ;
model=buildcnn();
weights=model.weights();
popsize=len(reshape(weights));
pop=createsolution(popsize);
gbest = gbestsolution(pop, IdFit, IdMinProb);
initialization;
while e in epoch do
    fitlist=[];
    for item in pop do
        fitlist ← item[IdFit];
    end
    for i in popsize do
        idx = roulettewheelselection(fitlist);
        lamda = alpha * uniform();
        newpos = pop[i][IdPos] + lamda × ((pop[idx][IdPos] + gbest[IdPos]) / 2
         - pop[i][IdPos]);
        temp = pop[i][IdPos] + normal(0, 1, popsize) × sigma;
        if uniform(0, 1, popsize) < pm then
            newpos = temp;
        end
        newpos = clip(newpos, lb, ub);
        fit = fitnesspos(newpos);
        pop[i] = [newpos, fit];
        if i % batch then
            gbest = updategbest(nextpop, IdMinProb, gbest);
        end
    end
    losstrain ← gbest[IdFit];
end
solution = gbest;
return gbest[IdPos], gbest[IdFit], losstrain;
```

Algorithm 5 uses the solution vector obtianed from the CNN model as a population. Then, the algorithm iterates over the number of epoch in order to train the CNN model. This algorithm, calculate the bowers' probability, which indicates the item

in the population or the solution set. Thereafter, it selects the bower and carries the mutation operation in different number of iteration using the roulette wheel. The global best solution is obtained from continually updated global best pool during the iteration on the population size.

3.4.5 Design of the Life Choice-Based Optimization CNN (LCBO-CNN)

The LCBO algorithm is motivated by observing the human being's life cycle and their ability for decision making to achieve their objectives while learning from fellow members Oyelade & Ezugwu (2021b); Khatri et al. (2020). LCBO is examined widely using 29 popular functions known as a benchmark functions which confirmed that it outperformed other optimization techniques. We used the LCBO to optimize the CNN training for lung cancer diagnosis and detection. The following steps explain the procedures of the LCBO, while the algorithmic design is presented in Algorithm listing 6.

1. Randomly initialize the human population.
2. Initialize the current chance (CC).
3. Compute and rank the fitness values.
4. while CC < No. of chances
   (a) For each search agent
      i. Generate a random number (r)
      ii. If r > 0.875
         - Update current search agent using

$$X_j^{'} = \sum_{k=1}^{n} [\frac{rand(k) * X_k}{n}]$$

           where n is the parameter that equal to the ceil of the square root of the human population considered to solve the problem, X_j indicates the j^{th} or the current search agent, and $X_j^{'}$ denotes the X_j will be updated when the $X_j^{'}$ has better fitness than X_j.

      iii. Else if r < 0.70
         - Update f1, f2 using

$$f1 = \frac{1-(CC-1)}{No.ofchances-1}$$
$$f2 = 1-f1$$

where $f1$ and $f2$ are vary linearly from 0 to 1 and 1 to 0, respectively.

- Update current search agent using

$$BestDiff = f1 \times c \times (X_1 - X_j)$$
$$BetterDiff = f2 \times c \times (X_{j-1} - X_j)$$
$$X_j^{'} = X_j + rand() \times BetterDiff + rand() \times BestDiff$$

where c is a constant, and X_{j-1} indicates the position of the search agent whose fitness was just better than the current agent till the previous iteration. X_1 is the best position of the agent that has been achieved till the previous iteration.

iv. Else

- Update the current search agnet using

$$X_j^{'} = X_{max} - (X_j - X_{min}) \times rand()$$

where X_{max} and X_{min} are the upper and lower bound values.

v. Evaluate fitness value of search agent If new fitness values better than previous

vi. Update agent position and fitness value

(b) Sort the population

(c) Increment CC by 1

5. Return population

Algorithm 6: Algorithm of the LCBO-CNN

```
Result: optimized CNN model
objfunc, lb, ub, batch, epoch, n1 solution, bestfit, losstrain initialize IdFit,
IdMinProb ;
model=buildcnn();
weights=model.weights();
popsize=len(reshape(weights));
pop=createsolution(popsize);
gbest = gbestsolution(pop, IdFit, IdMinProb);
initialization;
while e in epoch do
    wf = 2 × (1 - (e + 1) / epoch);
    for i in popsize do
        if i < n1 then
            for j in n1 do
                posnew= ← wf × uniform() × pop[j][IdPos];
                posnew = findmean(posnew);
            end
        else
            if n1 ≤ i < n2 then
                posnew = gbest[IdPos] + computestepsizebylevyflight(0.01, 1.5) ×
                  (gbest[IdPos] - pop[i][IdPos]);
            else
                betterdiff = wf × uniform() × (pop[i - 1][IdPos] - pop[i][IdPos]);
                bestdiff = (2 - wf) × uniform() × (pop[0][IdPos] - pop[i][IdPos]);
                posnew = pop[i][IdPos] + wf × uniform() × (normal() × betterdiff
                  + bestdiff);
            end
        end
        fit = fitnesspos(posnew);
        if fit < pop[i][IdFit] then
            pop[i] = [posnew, fit];
        else
            posnew = ub - (pop[i][IdPos] - lb) × uniform(lb, ub);
            fit = fitnesspos(posnew);
            pop[i] = [posnew, fit];
        end
        if i%batch then
            gbest = updategbest(nextpop,IdMinProb, gbest);
        end
    end
    losstrain ← gbest[IdFit];
end
solution = gbest;
return gbest[IdPos], gbest[IdFit], losstrain
```

In algorithm 6 above, the population is initialized from the solution vector calculated by the CNN model. Then, it uses the loop to obtain the loss function values at the end of the iterations.

3.4.6 Design of the Ebola Optimization Algorithm CNN (EOSA-CNN)

This algorithm is based on the propagation of Ebola virus disease (EVD) Oyelade & Ezugwu (2021c). The EOSA model updates the population using a dynamic mechanism for propagating through susceptible, infection, quarantine, recovered, and hospitalized operations for a better fit. Moreover, EOSA helps discover the best or worst candidate solution and gives intuitive results. It has been applied widely and investigated using about 50 benchmark functions Oyelade & Ezugwu (2021a); Jamil & Yang (2013). The following steps explain the procedures of the EOSA, while the algorithmic design is presented in Algorithm listing 7.

1. Initialize the vector and scalar quantities for the individuals and parameters, respectively. Individuals in the sets: Susceptible (S), Infected (I), Recovered (R), Dead (D), Vaccinated (V), Hospitalized (H), and Quarantine (Q) with their initial values.
2. Generate the index case (I_1) randomly from the susceptible individuals.
3. Specify the index case as the global best and current best, and calculate the fitness value of the index case.
4. While the number of iterations is not finished and there exists at least an infected individual, then
   (a) For each susceptible individual generates and updates their position based on their displacement. Note that the further an infected case is displaced, the more the infection number, so short displacement describes exploitation, otherwise exploration.
      i. Generate newly infected individuals (nI) based on (a).
      ii. Add the newly generated cases to I.
   (b) Calculate the number of individuals to be added to H, D, R, B, V, and Q utilizing their respective rates based on the size of I.
   (c) Update S and I base on nI.
   (d) Pick the current best from I and compare it with the global best.
   (e) If the requirement for the ending is not met, go back to step 6.
5. Return global best solution and all solutions.

The mathematical model representing the above steps can be shown based on the definition of the susceptible (S), Infected (I), Recovered (R), Dead (D), Vaccinated (V), Hospitalized (H), and Quarantine (Q). Meanwhile, the positions of each exposed individual can be calculated using

$$mI_i^{t+1} = mI_i^t + \rho M(I) \tag{3.5}$$

where ρ indicates the scale factor of displacement such individuals, mI_i^{t+1} and mI_i^t denote the updated and orignial position at time t, respectively. $t+1$ is the current time. $M(I)$ represent the movement rate made by individuals which can be computed using

$$M(I) = srate \times rand(0,1) + M(Ind_{best}) \tag{3.6}$$

$$M(S) = lrate \times rand(0,1) + M(Ind_{best}) \tag{3.7}$$

The exploration stage of EOSA described when the current position of the infected individual has moved beyond the normal neighborhood range *lrate*. While the exploitation phase of the EOSA algorithm supposes that the infected individual either remains within a distance of zero (0) or is displaced within a limit not exceeding *srate* when compared to its previous position.

To initialize the susceptible population, we generate a population using the random number distribution. The individual i can be generated using the following equation:

$$individual_i = L_i + rand(0,1) \times (U_i + L_i) \tag{3.8}$$

where U_i, L_i indicate the the upper and lower bounds for the i^{th} individual, $i = 1, 2, \ldots, N$.
The selection of the current best is carried out on the set of infected individuals in time t. The global best (*gBest*) can be computed using the following:

$$bestS = \begin{cases} gBest, & fitness(cBest) < fitness(gBest) \\ cBest, & fitness(cBest) \geq fitness(gBest) \end{cases} \tag{3.9}$$

where $bestS$, $gBest$, and $cBest$ indicates the best solution, global best solution, and current best solution at time t. The $fitness$ denote the objective function used for the problem.
A set of differential calculus used to update the susceptible (S), Infected (I), Recovered (R), Dead (D), Vaccinated (V), Hospitalized (H), Funeral (F), Exposed (E), and

Quarantine (Q) as follows

$$\frac{\partial S(t)}{\partial t} = -(\beta_1 I + \beta_3 D + \beta_4 R + \beta_2 (PE))S - (\tau S + \Gamma I) \tag{3.10}$$

$$\frac{\partial I(t)}{\partial t} = (\beta_1 I + \beta_3 D + \beta_4 R + \beta_2 (PE)\lambda)S - (\Gamma + \gamma)I - (\tau)S \tag{3.11}$$

$$\frac{\partial H(t)}{\partial t} = \alpha I - (\gamma + \varpi)H \tag{3.12}$$

$$\frac{\partial R(t)}{\partial t} = \gamma I - \Gamma R \tag{3.13}$$

$$\frac{\partial V(t)}{\partial t} = \gamma I - (\mu + \vartheta)V \tag{3.14}$$

$$\frac{\partial D(t)}{\partial t} = (\tau S + \Gamma I) - \delta D \tag{3.15}$$

$$\frac{\partial Q(t)}{\partial t} = (I - (\gamma R + \Gamma D)) - \xi Q \tag{3.16}$$

The equations (3.10 - 3.16) are scalar functions. Each function has one number which is represented as a float. We specify the rate of change of the population of susceptible individuals. Then, we implement it to the current size of the susceptible vector to get the number susceptible individuals at time t. This procedure is implemented to calculate the set of individuals in vectors I, H, R, V, D, and Q. Our study assumes the initial conditions S(0) = S0, I(0) = I0, R(0) = R0, D(0) = D0, P(0) = P0, and Q(0) = Q0 where our t follows after the epoch, and δ in equation 3.15 is for the burial rate. Equation 3.16 denotes the rate of quarantine of infected cases of Ebola.

Algorithm 7: Algorithm of the EOSA-CNN

Result: optimized CNN model

objfunc, lb, ub, batch, epoch, psize, evdincub;
solution, bestfit, losstrain;
S, E, I, H, R, V, Q, solution ← ∅ ;
initialize S finite set S = $\{ind_1, ind_2, \ldots, ind_n\}$;
S ← createSusceptibleIndvd(psize, S), Eq. 3.8;
time ← 0;
icase ← generatedIndexCase();
gbest, cbest ← icase;
while $e \leq epoch \wedge len(I) > 0$ **do**
    Q ← rand(0, Eq. 3.16 × I);
    fracI = I - Q;
    for $i \leftarrow 1$ **to** $len(fracI)$ **do**
        pos_i← moverate() using Eq. 3.5;
        d_i← rand();
        if $d_i > evdincub$ **then**
            neighborhood ← prob(pos_i)
            if $neighborhood < 0.5$ **then**
                tmp ← rand(0, Eq. 3.11 × I × srate);
            else
                tmp ← rand(0, Eq. 3.11 × I × lrate);
            end
            newI+ ← tmp ;
        end
        I+ ← newI ;
    end
    h ← rand(0, Eq. 3.12 × I), H+ ← h;
    r ← rand(0, Eq. 3.13 × I), R+ ← r;
    v ← rand(0, Eq. 3.14 × h), V+ ← v;
    d ← rand(0, Eq. 3.15 × I), D+ ← d;
    I+ ← I - add(r, d);
    S+ ← r;
    S- ← d;
    cbest = fitness(obj func, I);
    if $cbest > gbest$ **then**
        gbest = cbest;
        solution ← gbest;
    end
end
return gbest, sols;

In the algorithm 7, lines 1-7 indicate the initialization stage. In this case, not all the infected individuals can recruit newly infected individuals. Line 8 shows that some I are admitted into quarantine status so that the rest fraction of I infect S population. Moreover, in lines 10-24, new infections cases are generated from S and then added to I. The R, V, H, and V are derivable from I. However, lines 25-29 create individuals using corresponding equations of subgroups. In addition, the recovered and dead cases should be removed from I before the next iteration. The recovered instances will be added to S with the new cases to promote the idea of new births as presented on lines 29-31. Ultimately, the best solution is calculated. Also, the termination criteria are checked. In the case of satisfied criteria, the algorithm terminates afterwards; otherwise, return to line 7.

3.5 Experimentation Settings

In this study, due to the device specification limits and the time needed to carry the experiments was very large, five experiments were conducted to examine and explore the performance of the traditional CNN model independently and the proposed CNN using the metaheuristic optimization algorithms including GA, SBO, MVO, WOA, LCBO, and EOSA. All the experiment are carried out on a dell machine (Optiplex 5050) with the following specifications: Intel core i5, 7^{th} generation, 16GB memory, and 500GB hard drive (additional hardware device specification is shown in Figure 3.13). Table 3.1 shows the proposed CNN hyperparameter configuration.

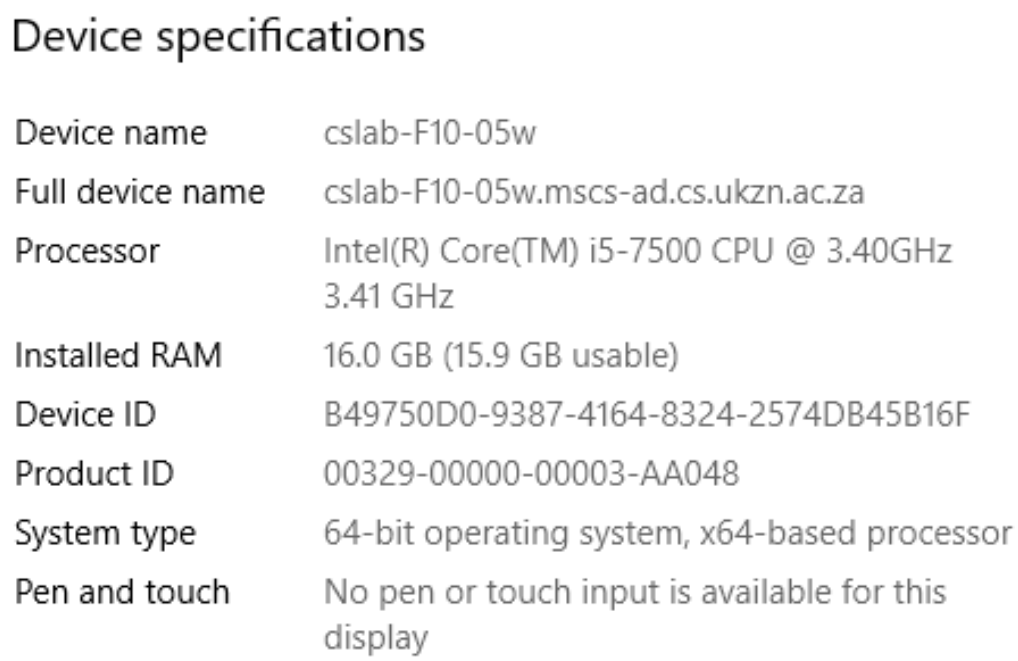


Figure 3.13: The specification of the device used to carry the experiments.

Table 3.1: Proposed CNN hyperparameter configuration

Parameter	Notatio	CNN architecture
Learning rate	α	0.001
Loss function	$I(x)$	categorical cross entropy
Epoch	e	5
Batch size	bs	32
Optimizer	θ_t	Adam
Kernel size/count	f/k	[3, 3]
Convolution layers	$conv$	[2conv-2conv]
Activation function	$\sum w_i b_i$	Relu
Pooling layers	P	[(2,2), (3,3)]
Padding/Stride	d/s	same / (1,1)

The input to the proposed CNN architectures is 258 × 258, which represents the pre-processed images of size 512 × 512. Moreover, Table 3.2 present the configuration of the metaheuristic algorithms for optimizing the proposed CNN model. All the methods share the same values of parameters such as the batch size and the number of epoch.

Table 3.2: Metaheuristic algorithms parameter configuration

Parameters	Value
GA-CNN Algorithm	
epoch	100
batch size	128
mutation percentage on the population	0.025
crossover percentage on the population	0,95
domain ranges (lower and upper)	[(1, 1)]
WOA-CNN Algorithm	
epoch	100
batch size	128
a	linearly decreased from 2 to 0
domain ranges (lower and upper)	[(1, 1)]
MVO-CNN Algorithm	
epoch	100
batch size	128
wep min-max	(1.0, 0.2)
domain ranges (lower and upper)	[(1, 1)]
SBO-CNN Algorithm	
epoch	100
batch size	128
alpha	[0.94]
z	[0.02]
mutation probability	[0.05]
domain ranges (lower and upper)	[(1, 1)]
LCBO-CNN Algorithm	
epoch	100
batch size	128
r1	2.35
domain ranges (lower and upper)	[(1, 1)]
EOSA-CNN Algorithm	
epoch	100
batch size	128
epxilon	0.001

3.6 Performance Metrics

In this work, we evsluation of the proposed method as compared to the five CNN metaheuristic algorithms and the traditional CNN models is based on seven performance metrics. These metrics are computed from the generic confusion matrix below.

Table 3.3: The confusion matrix

	True Class	
Predicted Class	Positive	Negative
Positive	True Positive (TP)	False Positive (FP)
Negative	False Negative (FN)	True Negative (TN)

1. **Accuracy** is the proportion of correctly classified cases to the total number of cases:
$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}.$$

2. **Cohen's Kappa** is a measure of agreement between the predictions and the true classes, controlling for the accuracy of a random classifier as measured by the expected accuracy:
$$Kappa = \frac{\text{Accuracy} - \text{Random Accuracy}}{1 - \text{Random Accuracy}}.$$

$$\begin{aligned}\text{Random Accuracy} &= \frac{\text{ActNegitave} \times \text{PredNegitave} + \text{ActPositive} \times \text{PredPositive}}{\text{Total} \times \text{Total}} \\ &= \frac{(TN + FP) \times (TN + FN) + (FN + TP) \times (FP + TP)}{(TP + TN + FP + FN) \times (TP + TN + FP + FN)}.\end{aligned}$$

3. **Specificity** is the proportion of true negatives which are predicted negative:
$$Specificity = \frac{TN}{(TN + FP)}.$$

4. **Sensitivity (Recall)** is the proportion of true positives which are predicted positive:
$$Sensitivity = Recall = \frac{TP}{(TP + FN)}.$$

5. **Precision** is the proportion of correctly predicted positive cases to the total

predicted positive cases:

$$Precision = \frac{TP}{TP + FP}$$

6. **F1 score** is the weighted mean of precision and recall:

$$\text{F1 Score} = \frac{2 * (Recall * Precision)}{(Recall + Precision)}$$

7. **Balanced Accuracy** is a metric used to evaluate the goodness of the performance of an algorithm. It is specifically helpful in the issue of imbalanced data. In other words, balanced accuracy is simply defined as the average of the sensitivity and specificity:

$$\text{Balanced Accuracy} = \frac{Sensitivity + Specificity}{2}$$

Chapter 4

Experimentation and Discussion of Results

The performance of the proposed hybrid algorithm EOSA-CNN was implemented using python. In addition, we compared the EOSA-CNN to other CNN solutions implemented on the same classification problem. Also, the proposed model performance was evaluated with five different hybrid algorithms, namely genetic algorithm (GA), Life choice-based optimization (LCBO), multi-Verse optimizer (MVO), satin bowerbird optimization (SBO), and whale optimization algorithm (WOA). The optimized CNN was trained using the dataset for classifying and detecting lung cancer. The trained model was then evaluated using a separate dataset for prediction. The experiment was carried out five times (see Tables 4.1 - 4.5). Thereafter, we computed the best, mean, standard deviation (SD), median, and worst of overall and class-based performance based on the GA-CNN, LCBO-CNN, MVO-CNN, SBO-CNN, WOA-CNN, and EOSA-CNN hybrid algorithms and as compared with the basic CNN architecture as shown in Tables 4.6 and 4.7.

Overall, the optimization process benefited the entire procedure and the results showed that EOSA-CNN outperformed other hybrid models by achieving superior performance accuracy of 0.82, compared to GA-CNN, LCBO-CNN, MVO-CNN, SBO-CNN, WOA-CNN which yielded 0.81, 0.81, 0.79, 0.81, 0.81, and 0.81, respectively. Although, the hybrid algorithms have close performances, however, all the hybrid outperformed the traditional CNN with no optimization applied which obtained an accuracy of 0.76.

Similarly, EOSA-CNN algorithm achieved superiority over other hybrid algorithms for Kappa, recall, F1 score, and specificity by obtaining 0.70, 0.83, 0.82, 0.82 and 0.98, respectively. These results indicated that applying the proposed EOSA-CNN hybrid

algorithm has improved the classification process, leading to better and accurate malignancy detection. Furthermore, we noted that the good performance of the hybrid algorithm for specificity metric showed that it could effectively detect true negative cases, reducing false negative reports.

We examined the performance of the EOSA-CNN algorithm on the three classes of labels seen on the samples drawn from the datasets. These results are also listed in Table 5, where the specificity, sensitivity, precision, recall, F1-score and balanced accuracy are computed and reported. In most cases, all the hybrid algorithms compete very closely with the proposed EOSA-CNN algorithm, while it is seen that it outperformed the traditional CNN in most of the metrics. Again, this confirms that EOSA-CNN successfully indented the features of each class and correctly classified them for a good performance. This further reinforces the need for the algorithm's usefulness in addressing the classification problem in the domain.

4.1 Overall and per class performance of the hybrid algorithms

Here, we presented the overall and per-class performance of the EOSA-CNN algorithm in the five experiments. These results illustrated that EOSA-CNN demonstrated superiority over all hybrid algorithms in the five experiments, with accuracies of 0.81, 0.81, 0.82, 0.82, 0.79. Moreover, EOSA-CNN scored the highest kappa values in the five runs, which indicates that EOSA-CNN has a good agreement between the prediction and the actual classes compared to other hybrid algorithms. Also, we noticed that the excellent performance of EOSA-CNN in terms of specificity metric showed that it could effectively detect the actual negative cases (see Tables 4.1 - 4.5).

Table 4.1: The overall and per class performance of the GA-CNN, LCBO-CNN, MVO-CNN, SBO-CNN, WOA-CNN, and EOSA-CNN hybrid algorithms and as compared with the basic CNN architecture (first run)

Measure/ Methods	GA-CNN	LCBO-CNN	MVO-CNN	SBO-CNN	WOA-CNN	EOSA-CNN	CNN
Overall Performance							
Accuracy	0.74	0.76	0.75	0.75	0.73	**0.81**	0.72
Cohens kappa	0.59	0.59	0.59	0.59	0.56	**0.67**	0.54
Percision	0.81	0.79	0.81	0.79	0.79	**0.81**	0.78
Recall	0.74	0.76	0.75	0.75	0.73	**0.81**	0.72
F1 score	0.77	0.76	0.77	0.76	0.75	**0.81**	0.74
Specificity	0.7	0.94	0.72	0.88	0.64	**0.92**	0.68
Sensitivity	0.66	0.2	0.43	0.33	0.37	**0.13**	0.4
Per Class Performance							
Sensitivity							
Normal	0.6635	0.9327	0.7212	0.8654	0.6058	**0.9231**	0.6538
Benign	0.63333	0.2	0.43333	0.3333	0.3333	**0.13333**	0.4
Malignant	0.8286	0.7429	0.8357	0.7571	0.9071	**0.8643**	0.8286
Specificity							
Normal	0.8765	0.7059	0.8176	0.7412	0.8824	**0.7941**	0.8176
Benign	0.82377	0.93443	0.84426	0.9098	0.8115	**0.92623**	0.8238
Malignant	0.9552	0.9925	1	0.9851	0.9403	**1**	0.9701
Precision							
Normal	0.7667	0.6599	0.7075	0.6716	0.759	**0.7328**	0.6869
Benign	0.30645	0.27273	0.2549	0.3125	0.1786	**0.18182**	0.2182
Malignant	0.9508	0.9905	1	0.9815	0.9407	**1**	0.9667
Recall							
Normal	0.6635	0.9327	0.7212	0.8654	0.6058	**0.9231**	0.6538
Benign	0.63333	0.2	0.43333	0.3333	0.3333	**0.13333**	0.4
Malignant	0.8286	0.7429	0.8357	0.7571	0.9071	**0.8643**	0.8286
F1 score							
Normal	0.7113	0.7729	0.7143	0.7563	0.6738	**0.817**	0.67
Benign	0.41304	0.23077	0.32099	0.3226	0.2326	**0.15385**	0.2824
Malignant	0.8855	0.849	0.9105	0.8548	0.9236	**0.9272**	0.8923
Balanced Accuracy							
Normal	0.77	0.8193	0.7694	0.8033	0.7441	**0.8586**	0.7357
Benign	0.72855	0.56721	0.6388	0.6216	0.5724	**0.52978**	0.6119
Malignant	0.8919	0.8677	0.9179	0.8711	0.9237	**0.9321**	0.8994

Table 4.1 above, describe the overall and the per-class result of the first experiment, we can see clearly that our proposed model achieved the highest performance (in bold) compared to the other models in terms of all the metrics.

Table 4.2: The overall and per class performance of the algorithms (second run)

Measure/ Methods	GA-CNN	LCBO-CNN	MVO-CNN	SBO-CNN	WOA-CNN	EOSA-CNN	CNN
Overall Performance							
Accuracy	0.77	0.78	0.78	0.8	0.8	**0.81**	0.76
Cohens kappa	0.62	0.64	0.63	0.67	0.66	**0.68**	0.6
Percision	0.78	0.81	0.8	0.86	0.83	**0.82**	0.81
Recall	0.77	0.78	0.78	0.8	0.8	**0.81**	0.76
F1 score	0.77	0.8	0.78	0.82	0.81	**0.82**	0.78
Specificity	0.9	0.78	0.9	0.73	0.78	**0.85**	0.73
Sensitivity	0.3	0.32	0.37	0.53	0.41	**0.35**	0.53
Per Class Performance							
Sensitivity							
Normal	0.8654	0.7692	0.8654	0.7308	0.75	**0.8269**	0.7115
Benign	0.26667	0.3	0.3333	0.53333	0.4	**0.3**	0.53333
Malignant	0.8143	0.8929	0.8143	0.9071	0.9214	**0.9071**	0.8357
Specificity							
Normal	0.7824	0.8588	0.8059	0.9059	0.8824	**0.8882**	0.8412
Benign	0.92623	0.86475	0.918	0.84016	0.877	**0.89344**	0.84836
Malignant	0.9478	0.9776	0.9478	1	0.9627	**0.9478**	0.9776
Precision							
Normal	0.7087	0.7692	0.7317	0.8261	0.7959	**0.819**	0.7327
Benign	0.30769	0.21429	0.3333	0.29091	0.2857	**0.25714**	0.30189
Malignant	0.9421	0.9766	0.9421	1	0.9627	**0.9478**	0.975
Recall							
Normal	0.8654	0.7692	0.8654	0.7308	0.75	**0.8269**	0.7115
Benign	0.26667	0.3	0.3333	0.53333	0.4	**0.3**	0.53333
Malignant	0.8143	0.8929	0.8143	0.9071	0.9214	**0.9071**	0.8357
F1 score							
Normal	0.7792	0.7692	0.793	0.7755	0.7723	**0.823**	0.722
Benign	0.28571	0.25	0.3333	0.37647	0.3333	**0.27692**	0.38554
Malignant	0.8736	0.9328	0.8736	0.9513	0.9416	**0.927**	0.9
Balanced Accuracy							
Normal	0.8239	0.814	0.8356	0.8183	0.8162	**0.8576**	0.7764
Benign	0.59645	0.58238	0.6257	0.68675	0.6385	**0.59672**	0.69085
Malignant	0.881	0.9352	0.881	0.9536	0.9421	**0.9275**	0.9067

Table 4.2 describe the overall and the per-class result of the second experiment, we notice clearly that our proposed model scored the highest performance (in bold) compared to the other models in terms of all the metrics.

Table 4.3: The overall and per class performance of the algorithms (third run)

Measure/ Methods	GA-CNN	LCBO-CNN	MVO-CNN	SBO-CNN	WOA-CNN	EOSA-CNN	CNN
Overall Performance							
Accuracy	0.81	0.81	0.79	0.81	0.81	**0.82**	0.73
Cohens kappa	0.67	0.68	0.66	0.67	0.67	**0.68**	0.57
Percision	0.83	0.84	0.85	0.81	0.81	**0.82**	0.8
Recall	0.81	0.81	0.79	0.81	0.81	**0.82**	0.73
F1 score	0.81	0.81	0.81	0.81	0.81	**0.82**	0.76
Specificity	0.82	0.94	0.82	0.98	0.93	**0.96**	0.68
Sensitivity	0.37	0.37	0.67	0.18	0.23	**0.15**	0.53
Per Class Performance							
Sensitivity							
Normal	0.8077	0.9368	0.8077	0.9712	0.9135	**0.9038**	0.6635
Benign	0.36667	0.36667	0.66667	0.16667	0.23333	**0.13333**	0.53333
Malignant	0.9	0.8143	0.8	0.8286	0.8571	**0.9071**	0.8286
Specificity							
Normal	0.8471	0.7941	0.8882	0.7529	0.7882	**0.7941**	0.8353
Benign	0.89754	0.93191	0.84836	0.97131	0.94262	**0.97951**	0.82787
Malignant	0.9851	1	0.9851	0.9776	0.9851	**0.9328**	0.9776
Precision							
Normal	0.7636	0.7177	0.8155	0.7063	0.7252	**0.7287**	0.7113
Benign	0.30556	0.40741	0.35088	0.41667	0.33333	**0.44444**	0.27586
Malignant	0.9844	1	0.9825	0.9748	0.9836	**0.9338**	0.9748
Recall							
Normal	0.8077	0.9368	0.8077	0.9712	0.9135	**0.9038**	0.6635
Benign	0.36667	0.36667	0.66667	0.16667	0.23333	**0.13333**	0.53333
Malignant	0.9	0.8143	0.8	0.8286	0.8571	**0.9071**	0.8286
F1 score							
Normal	0.785	0.8128	0.8116	0.8178	0.8085	**0.8069**	0.6866
Benign	0.33333	0.38596	0.45977	0.2381	0.27451	**0.20513**	0.36364
Malignant	0.9403	0.8976	0.8819	0.8958	0.916	**0.9203**	0.8958
Balanced Accuracy							
Normal	0.8274	0.8655	0.848	0.862	0.8508	**0.849**	0.7494
Benign	0.6321	0.64929	0.75751	0.56899	0.58798	**0.55642**	0.6806
Malignant	0.9425	0.9071	0.8925	0.9031	0.9211	**0.92**	0.9031

Table 4.3 describe the overall and the per-class result of the third experiment, our proposed model outperformed the other models (in bold) in terms of all the metrics.

Table 4.4: The overall and per class performance of the algorithms (fourth run)

Measure/ Methods	GA-CNN	LCBO-CNN	MVO-CNN	SBO-CNN	WOA-CNN	EOSA-CNN	CNN
Overall Performance							
Accuracy	0.77	0.77	0.76	0.77	0.77	**0.82**	0.75
Cohens kappa	0.63	0.61	0.59	0.62	0.63	**0.7**	0.6
Percision	0.84	0.81	0.79	0.81	0.82	**0.83**	0.81
Recall	0.77	0.77	0.76	0.77	0.77	**0.82**	0.75
F1 score	0.8	0.78	0.77	0.79	0.79	**0.82**	0.78
Specificity	0.73	0.86	0.82	0.76	0.75	**0.98**	0.7
Sensitivity	0.57	0.45	0.33	0.43	0.41	**0.38**	0.53
Performance per class							
Sensitivity							
Normal	0.7212	0.8558	0.7885	0.75	0.7404	**0.9231**	0.6827
Benign	0.53333	0.43333	0.3333	0.43333	0.4	**0.36667**	0.53333
Malignant	0.8643	0.7714	0.8214	0.8571	0.8786	**0.85**	0.85
Specificity							
Normal	0.8941	0.7765	0.7882	0.8235	0.8588	**0.8294**	0.8529
Benign	0.83197	0.90164	0.8893	0.86885	0.8525	**0.95082**	0.83197
Malignant	0.9776	0.9851	0.9701	0.9925	0.9851	**0.9478**	0.9851
Precision							
Normal	0.8065	0.7008	0.6949	0.7222	0.7624	**0.768**	0.7396
Benign	0.2807	0.35135	0.2703	0.28889	0.25	**0.47826**	0.2807
Malignant	0.9758	0.9818	0.9664	0.9917	0.984	**0.9444**	0.9835
Recall							
Normal	0.7212	0.8558	0.7885	0.75	0.7404	**0.9231**	0.6827
Benign	0.53333	0.43333	0.3333	0.43333	0.4	**0.36667**	0.53333
Malignant	0.8643	0.7714	0.8214	0.8571	0.8786	**0.85**	0.85
F1 score							
Normal	0.7614	0.7706	0.7387	0.7358	0.7512	**0.8384**	0.71
Benign	0.36782	0.38806	0.2985	0.34667	0.3077	**0.41509**	0.36782
Malignant	0.9167	0.864	0.888	0.9195	0.9283	**0.8947**	0.9119
Balanced Accuracy							
Normal	0.8076	0.8161	0.7883	0.7868	0.7996	**0.8762**	0.7678
Benign	0.68265	0.66749	0.6113	0.65109	0.6262	**0.65874**	0.68265
Malignant	0.9209	0.8783	0.8958	0.9248	0.9318	**0.8989**	0.9175

Table 4.4 describe the overall and the per-class result of the fourth experiment, we notice clearly that our proposed model achieved the highest performance (in bold) compared to the other models in terms of all the metrics.

Table 4.5: The overall and per class performance of the algorithms (fifth run)

Measure/ Methods	GA-CNN	LCBO-CNN	MVO-CNN	SBO-CNN	WOA-CNN	EOSA-CNN	CNN
Overall Performance							
Accuracy	0.76	0.78	0.78	0.75	0.78	**0.79**	0.73
Cohens kappa	0.6	0.65	0.63	0.58	0.64	**0.65**	0.56
Precision	0.79	0.87	0.79	0.78	0.81	**0.81**	0.79
Recall	0.76	0.78	0.78	0.75	0.78	**0.79**	0.73
F1 score	0.78	0.8	0.78	0.76	0.79	**0.79**	0.75
Specificity	0.76	0.62	0.89	0.77	0.94	**0.92**	0.67
Sensitivity	0.35	0.79	0.18	0.28	0.37	**0.27**	0.43
Per Class Performance							
Sensitivity							
Normal	0.7308	0.5962	0.875	0.7596	0.9135	**0.9231**	0.6538
Benign	0.3	0.76667	0.16667	0.2667	0.36667	**0.26667**	0.43333
Malignant	0.8857	0.9214	0.8429	0.8429	0.7786	**0.8071**	0.85
Specificity							
Normal	0.8647	0.9588	0.7882	0.7882	0.7588	**0.7588**	0.8176
Benign	0.86066	0.80328	0.91803	0.877	0.93852	**0.93443**	0.83607
Malignant	0.9403	0.9627	0.9701	0.9776	0.9776	**1**	0.9776
Precision							
Normal	0.7677	0.8986	0.7165	0.687	0.6985	**0.7007**	0.6869
Benign	0.2093	0.32394	0.2	0.2105	0.42308	**0.33333**	0.24528
Malignant	0.9394	0.9627	0.9672	0.9752	0.9732	**1**	0.9754
Recall							
Normal	0.7308	0.5962	0.875	0.7596	0.9135	**0.9231**	0.6538
Benign	0.3	0.76667	0.16667	0.2667	0.36667	**0.26667**	0.43333
Malignant	0.8857	0.9214	0.8429	0.8429	0.7786	**0.8071**	0.85
F1 score							
Normal	0.7488	0.7168	0.7879	0.7215	0.7917	**0.7967**	0.67
Benign	0.24658	0.45545	0.18182	0.2353	0.39286	**0.2963**	0.31325
Malignant	0.9118	0.9416	0.9008	0.9042	0.8651	**0.8933**	0.9084
Balanced Accuracy							
Normal	0.7977	0.7775	0.8316	0.7739	0.8361	**0.841**	0.7357
Benign	0.58033	0.78497	0.54235	0.5719	0.6526	**0.60055**	0.6347
Malignant	0.913	0.9421	0.9065	0.9102	0.8781	**0.9036**	0.9138

Table 4.5 describe the overall and the per-class result of the fifth experiment, we notice clearly that our proposed model achieved the highest performance (in bold) compared to the other models in terms of all the metrics.

4.2 Performance metrics of the overall and per class performance of the algorithms

To provide a more detailed report on the performance of the hybrid algorithms when compared with the EOSA-CNN algorithm, and then with the traditional CNN, we computed the best, mean, standard deviation, median, and worst values obtained in all cases in Tables 4.6 and 4.7. We observed from Table 4.6 the best accuracy obtained for GA-CNN, LCBO-CNN, MVO-CNN, SBO-CNN, WOA-CNN, and EOSA-CNN are 0.81, 0.81, 0.79, 0.81, 0.81, and 0.82, respectively. This indicates that EOSA-CNN achieved a better performance when compared with other hybrid algorithms. Also, EOSA-CNN increases the classification performance with 0.06 compared to the traditional CNN. Due to data imbalance, it seems that overall accuracy is not a suitable fitness measure. However, we can use recall as a different performance measures. In this case EOSA-CNN exhibited good performance in terms of recall compared to other hybrid algorithms and surpassed the traditional CNN as can be seen in the following GA-CNN, LCBO-CNN, MVO-CNN, SBO-CNN, WOA-CNN, EOSA-CNN, and CNN yielded 0.81, 0.81, 0.79, 0.81, 0.81, 0.82, and 0.76, respectively. Furthermore, GA-CNN, LCBO-CNN, MVO-CNN, SBO-CNN, WOA-CNN, EOSA-CNN and CNN reported for specificity are 0.90, 0.94, 0.90, 0.98, 0.94, 0.98, and 0.73 respectively. For f1score, GA-CNN, LCBO-CNN, MVO-CNN, SBO-CNN, WOA-CNN, EOSA-CNN and CNN scored 0.81, 0.81, 0.81, 0.82, 0.81, 0.82, and 0.78, respectively.

Table 4.6: Best, mean, standard deviation, median, and worst of overall performance based on the five runs

Measure/ Methods		GA-CNN	LCBO-CNN	MVO-CNN	SBO-CNN	WOA-CNN	EOSA-CNN	CNN
Accuracy	**Best**	0,81	0.81	0.79	0.81	0.81	**0.82**	0.76
	Mean	0.77	0.78	0.772	0.776	0.778	**0.81**	0.738
	SD	0.025495	0.018708	0.016432	0.027928	0.031145	**0.012247**	0.016432
	Median	0.77	0.78	0.78	0.77	0.78	**0.81**	0.73
	Worst	0.74	0.76	0.75	0.75	0.73	**0.79**	0.72
Kappa	**Best**	0.67	0.68	0.66	0.67	0.67	**0.7**	0.6
	Mean	0.622	0.634	0.62	0.626	0.632	**0.676**	0.574
	SD	0.031145	0.035071	0.03	0.042778	0.043243	**0.018166**	0.026077
	Median	0.62	0.64	0.63	0.62	0.64	**0.68**	0.57
	Worst	0.59	0.59	0.59	0.58	0.56	**0.65**	0.54
Precision	**Best**	0.84	0.87	0.85	0.86	0.83	**0.83**	0.81
	Mean	0.81	0.824	0.808	0.81	0.812	**0.818**	0.798
	SD	0.025495	0.031305	0.0249	0.030822	0.014832	**0.008367**	0.013038
	Median	0.81	0.81	0.8	0.81	0.81	**0.82**	0.8
	Worst	0.78	0.79	0.79	0.78	0.79	**0.81**	0.78
Recall	**Best**	0.81	0.81	0.79	0.81	0.81	**0.82**	0.76
	Mean	0.77	0.78	0.772	0.776	0.778	**0.81**	0.738
	SD	0.025495	0.018708	0.016432	0.027928	0.031145	**0.012247**	0.016432
	Median	0.77	0.78	0.78	0.77	0.78	**0.81**	0.73
	Worst	0.74	0.76	0.75	0.75	0.73	**0.79**	0.72
F1 score	**Best**	0.81	0.81	0.81	0.82	0.81	**0.82**	0.78
	Mean	0.786	0.79	0.782	0.788	0.79	**0.812**	0.762
	SD	0.018166	0.02	0.016432	0.027749	0.024495	**0.013038**	0.017889
	Median	0.78	0.8	0.78	0.79	0.79	**0.82**	0.76
	Worst	0.77	0.76	0.77	0.76	0.75	**0.79**	0.74
Specificity	**Best**	0.9	0.94	0.9	0.98	0.94	**0.98**	0.73
	Mean	0.782	0.828	0.83	0.824	0.808	**0.926**	0.692
	SD	0.079498	0.133866	0.072111	0.104067	0.127161	**0.0498**	0.023875
	Median	0.76	0.86	0.82	0.77	0.78	**0.92**	0.68
	Worst	0.7	0.62	0.72	0.73	0.64	**0.85**	0.67
Sensitivity	**Best**	0.66	0.79	0.67	0.53	0.41	**0.38**	0.53
	Mean	0.45	0.426	0.396	0.35	0.358	**0.256**	0.484
	SD	0.156045	0.222778	0.17883	0.135093	0.074297	**0.11349**	0.063875
	Median	0.37	0.37	0.37	0.33	0.37	**0.27**	0.53
	Worst	0.3	0.2	0.18	0.18	0.23	**0.13**	0.4

In Table 4.6 above, the best, mean, standard deviation, median, and worst result of the overall performance were calculated. EOSA-CNN outperformed the other models in terms of average of the five experiments. Also, we noticed that EOSA-CNN was able to detect the malignancy cases with high accuracy. The result of the EOSA-CNN are highlighted in bold.

In Table 4.7, we calculated the best, mean, standard deviation, median, and worst performance for class labels seen in the samples from the dataset. We noticed that all hybrid algorithms and the traditional CNN were highly accurate in classifying

the malignancy class in terms of all the metrics. Furthermore, the hybrid algorithms achieved high performance compared to the traditional CNN, while EOSA-CNN outperform all the hybrid algorithms. In terms of sensitivity GA-CNN, LCBO-CNN, MVO-CNN, SBO-CNN, and WOA-CNN scored 0.90, 0.9214, 0.8429, 0.9071, and 0.9214 respectively. Where as EOSA-CNN achieved 0.9071 which is better compared to the CNN with sensitivity of 0.85. In contrast, for specificity GA-CNN, LCBO-CNN, MVO-CNN, SBO-CNN, and WOA-CNN attained 0.9851, 1, 1, 1, 0.9851, respectively. While EOSA-CNN, obtained 1 which is better compared to the traditional CNN that is achieved 0.9851. Overall, this results explain that EOSA-CNN has better performance compared to the other hybrids and traditional CNN algorithms. Moreover, since the dataset is imbalanced we have investigated other metrics for more confirmation, this metrics are precision, recall, f1score, and balanced accuracy. Overall, we observed that EOSA-CNN outperformed the other hybrid algorithms and the traditional CNN. Consequently, a good competitive performance is seen for the classification accuracy of all hybrid algorithms, with the basic CNN architecture lagging behind.

Table 4.7: Best, mean, standard deviation, median, and worst of per class performance based on the five runs.

Measure/ Methods		GA	LCBO	MVO	SBO	WOA	EOSA	CNN
				Sensitivity				
Normal	**Best**	0.8654	0.9368	0.875	0.9712	0.9135	**0.9231**	0.7115
	Mean	0.75772	0.81814	0.81156	0.8154	0.78464	**0.9**	0.67306
	SD	0.0791	0.14166	0.06252	0.10164	0.13072	**0.04171**	0.02451
	Median	0.7308	0.8558	0.8077	0.7596	0.75	**0.9231**	0.6635
	Worst	0.6635	0.5962	0.7212	0.7308	0.6058	**0.8269**	0.6538
Benign	**Best**	0.63333	0.76667	0.66667	0.53333	0.4	**0.36667**	0.53333
	Mean	0.42	0.41333	0.38665	0.34667	0.34666	**0.24**	0.48666
	SD	0.15741	0.21551	0.18349	0.14259	0.06912	**0.10382**	0.06498
	Median	0.36667	0.36667	0.3333	0.3333	0.36667	**0.26667**	0.53333
	Worst	0.26667	0.2	0.16667	0.16667	**0.23333**	0.13333	0.4
Malignant	**Best**	0.9	0.9214	0.8429	0.9071	0.9214	**0.9071**	0.85
	Mean	0.85858	0.82858	0.82286	0.83856	0.86856	**0.86712**	0.83858
	SD	0.03655	0.07676	0.01706	0.0543	0.05613	**0.04213**	0.01082
	Median	0.8643	0.8143	0.8214	0.8429	0.8786	**0.8643**	0.8357

Continued on next page

Table 4.7 – continued from previous page

Measure/ Methods		GA	LCBO	MVO	SBO	WOA	EOSA	CNN
	Worst	0.8143	0.7429	0.8	0.7571	0.7786	**0.8071**	0.8286
					Specificity			
Normal	**Best**	0.8941	0.9588	0.8882	0.9059	0.8824	**0.8882**	0.8529
	Mean	0.85296	0.81882	0.81762	0.80234	0.83412	**0.81292**	0.83292
	SD	0.043	0.09532	0.04138	0.06625	0.05712	**0.04893**	0.01535
	Median	0.8647	0.7941	0.8059	0.7882	0.8588	**0.7941**	0.8353
	Worst	0.7824	0.7059	0.7882	0.7412	0.7588	**0.7588**	0.8176
Benign	**Best**	0.92623	0.93443	0.91803	0.97131	0.94262	**0.97951**	0.84836
	Mean	0.86803	0.8872	0.88359	0.89342	0.88443	**0.93689**	0.83361
	SD	0.04351	0.05474	0.03602	0.05011	0.05636	**0.03169**	0.00943
	Median	0.86066	0.90164	0.8893	0.877	0.877	**0.93443**	0.83197
	Worst	0.82377	0.80328	0.84426	0.84016	0.8115	**0.89344**	0.8238
Malignant	**Best**	0.9851	1	1	1	0.9851	**1**	0.9851
	Mean	0.9612	0.98358	0.97462	0.98656	0.97016	**0.96568**	0.9776
	SD	0.01932	0.01435	0.01946	0.00973	0.01903	**0.03192**	0.0053
	Median	0.9552	0.9851	0.9701	0.9851	0.9776	**0.9478**	0.9776
	Worst	0.9403	0.9627	0.9478	0.9776	0.9403	**0.9328**	0.9701
					Precision			
Normal	**Best**	0.8065	0.8986	0.8155	0.8261	0.7959	**0.819**	0.7396
	Mean	0.76264	0.74924	0.73322	0.72264	0.7482	**0.74984**	0.71148
	SD	0.03492	0.09224	0.04791	0.06092	0.03739	**0.04546**	0.02475
	Median	0.7667	0.7177	0.7165	0.7063	0.759	**0.7328**	0.7113
	Worst	0.7087	0.6599	0.6949	0.6716	0.6985	**0.7007**	0.6869
Benign	**Best**	0.30769	0.40741	0.35088	0.41667	0.42308	**0.47826**	0.30189
	Mean	0.28194	0.31394	0.28188	0.30389	0.29414	**0.339**	0.26439
	SD	0.04213	0.07393	0.06118	0.07401	0.09156	**0.12445**	0.0328
	Median	0.30556	0.32394	0.2703	0.29091	0.2857	**0.33333**	0.27586
	Worst	0.2093	0.21429	0.2	0.2105	0.1786	**0.18182**	0.2182
Malignant	**Best**	0.9844	1	1	1	0.984	**1**	0.9835
	Mean	0.9585	0.98232	0.97164	0.98464	0.96884	**0.9652**	0.97508
	SD	0.02039	0.01412	0.02146	0.01097	0.018	**0.03218**	0.00594
	Median	0.9508	0.9818	0.9672	0.9815	0.9732	**0.9478**	0.975

Continued on next page

Table 4.7 – continued from previous page

Measure/ Methods		GA	LCBO	MVO	SBO	WOA	EOSA	CNN
	Worst	0.9394	0.9627	0.9421	0.9748	0.9407	**0.9338**	0.9667
					Recall			
Normal	**Best**	0.8654	0.9368	0.875	0.9712	0.9135	**0.9231**	0.7115
	Mean	0.75772	0.81814	0.81156	0.8154	0.78464	**0.9**	0.67306
	SD	0.0791	0.14166	0.06252	0.10164	0.13072	**0.04171**	0.02451
	Median	0.7308	0.8558	0.8077	0.7596	0.75	**0.9231**	0.6635
	Worst	0.6635	0.5962	0.7212	0.7308	0.6058	**0.8269**	0.6538
Benign	**Best**	0.63333	0.76667	0.66667	0.53333	0.4	**0.36667**	0.53333
	Mean	0.42	0.41333	0.38665	0.34667	0.34666	**0.24**	0.48666
	SD	0.15741	0.21551	0.18349	0.14259	0.06912	**0.10382**	0.06498
	Median	0.36667	0.36667	0.3333	0.3333	0.36667	**0.26667**	0.53333
	Worst	0.26667	0.2	0.16667	0.16667	0.23333	**0.13333**	0.4
Malignant	**Best**	0.9	0.9214	0.8429	0.9071	0.9214	0**.9071**	0.85
	Mean	0.85858	0.82858	0.82286	0.83856	0.86856	**0.86712**	0.83858
	SD	0.03655	0.07676	0.01706	0.0543	0.05613	**0.04213**	0.01082
	Median	0.8643	0.8143	0.8214	0.8429	0.8786	**0.8643**	0.8357
	Worst	0.8143	0.7429	0.8	0.7571	0.7786	**0.8071**	0.8286
					F1 score			
Normal	**Best**	0.785	0.8128	0.8116	0.8178	0.8085	**0.8384**	0.722
	Mean	0.75714	0.76846	0.7691	0.76138	0.7595	**0.8164**	0.69172
	SD	0.02938	0.03413	0.0408	0.03759	0.05248	**0.01586**	0.02356
	Median	0.7614	0.7706	0.7879	0.7563	0.7723	**0.817**	0.6866
	Worst	0.7113	0.7168	0.7143	0.7215	0.6738	**0.7967**	0.67
Benign	**Best**	0.41304	0.45545	0.45977	0.37647	0.39286	**0.41509**	0.38554
	Mean	0.3293	0.34205	0.31888	0.30383	0.30819	**0.26946**	0.34253
	SD	0.06568	0.09716	0.09907	0.06419	0.06051	**0.09936**	0.04303
	Median	0.33333	0.38596	0.32099	0.3226	0.3077	**0.27692**	0.36364
	Worst	0.24658	0.23077	0.18182	0.2353	0.2326	**0.15385**	0.2824
Malignant	**Best**	0.9403	0.9416	0.9105	0.9513	0.9416	**0.9272**	0.9119
	Mean	0.90558	0.897	0.89096	0.90512	0.91492	**0.9125**	0.90168
	SD	0.02643	0.04082	0.01476	0.03521	0.02937	**0.01712**	0.00829
	Median	0.9118	0.8976	0.888	0.9042	0.9236	**0.9203**	0.9

Continued on next page

Table 4.7 – continued from previous page

Measure/ Methods		GA	LCBO	MVO	SBO	WOA	EOSA	CNN
	Worst	0.8736	0.849	0.8736	0.8548	0.8651	**0.8933**	0.8923
					Balanced Accuracy			
Normal	**Best**	0.8274	0.8655	0.848	0.862	0.8508	**0.8762**	0.7764
	Mean	0.80532	0.81848	0.81458	0.80886	0.80936	**0.85648**	0.753
	SD	0.02315	0.03129	0.03384	0.03411	0.04133	**0.01314**	0.01856
	Median	0.8076	0.8161	0.8316	0.8033	0.8162	**0.8576**	0.7494
	Worst	0.77	0.7775	0.7694	0.7739	0.7441	**0.841**	0.7357
Benign	**Best**	0.72855	0.78497	0.75751	0.68675	0.6526	**0.65874**	0.69085
	Mean	0.64402	0.65027	0.63513	0.62007	0.61554	**0.58844**	0.66014
	SD	0.06144	0.08653	0.07787	0.05084	0.03404	**0.04905**	0.03479
	Median	0.6321	0.64929	0.6257	0.6216	0.6262	**0.59672**	0.6806
	Worst	0.58033	0.56721	0.54235	0.56899	0.5724	**0.52978**	0.6119
Malignant	**Best**	0.9425	0.9421	0.9179	0.9536	0.9421	**0.9321**	0.9175
	Mean	0.90986	0.90608	0.89874	0.91256	0.91936	**0.91642**	0.9081
	SD	0.02425	0.03313	0.01405	0.0302	0.02448	**0.0146**	0.00747
	Median	0.913	0.9071	0.8958	0.9102	0.9237	**0.92**	0.9067
	Worst	0.881	0.8677	0.881	0.8711	0.8781	**0.8989**	0.8994

Figure 4.1 shows the confusion matrix plot for all hybrid algorithms with respect to all the class labels observed in the dataset. The classification accuracy of all classes is indicated for each plot of the confusion matrix to give an accurate report on their performances. Taking the case of EOSA-CNN as an example, we see that 90% of all cases with normal labels were correctly identified and over 86% of cases labelled as malignant were correctly identified by the hybrid algorithm proposed in this study. This is contrary to what is reported for the traditional CNN, where only 67.31% of samples with normal labels were correctly identified while about 83% of those with malignancy were correctly identified. This, therefore, reinforces the impact of the hybrid algorithm proposed in this study since it was able to improve classification accuracy.

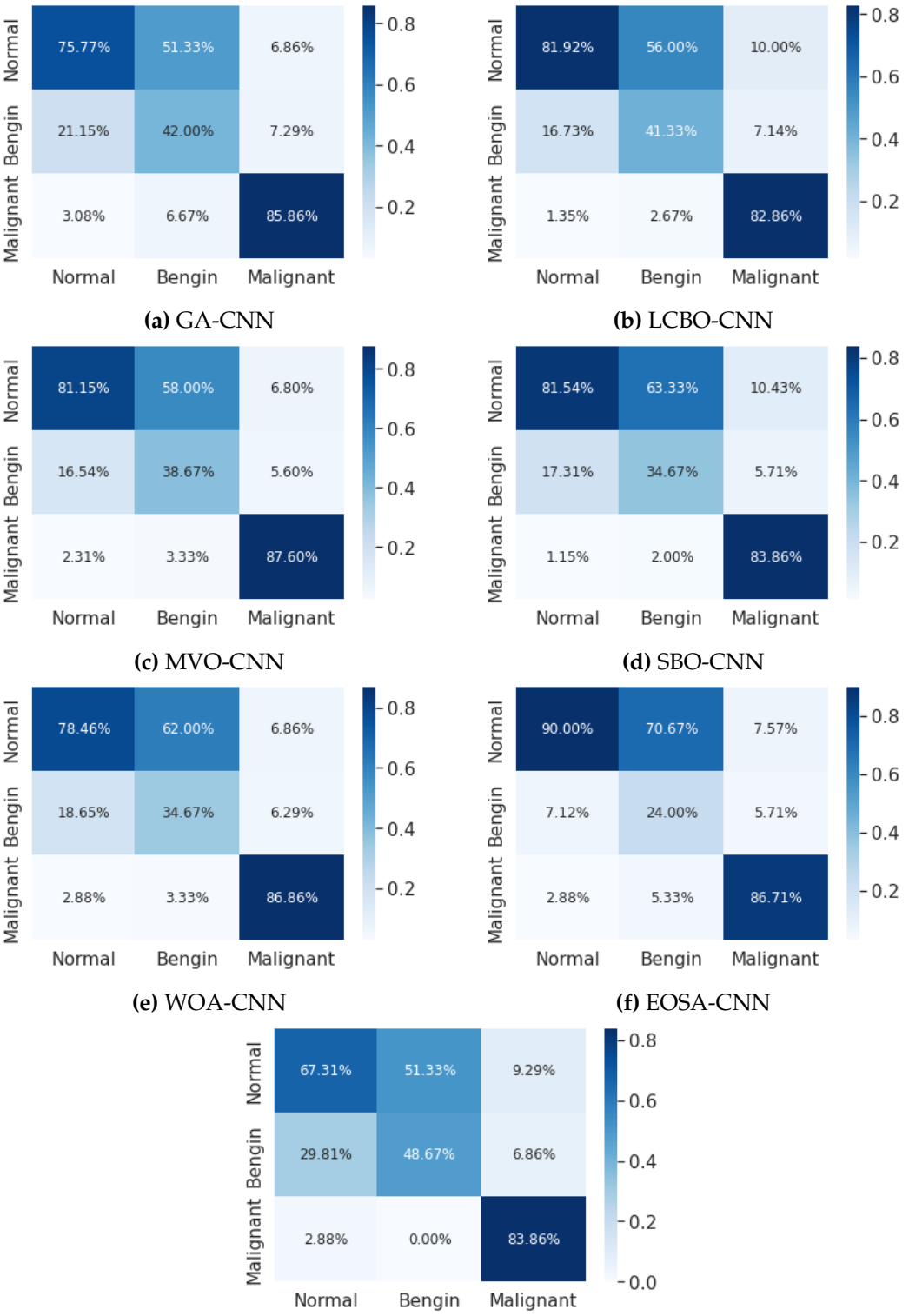


Figure 4.1: Overlapped confusion matrix for all hybrid algorithms with CNN.

For more confirmation, we plotted the validation epoch, as can be seen in Figure

4.2. These results indicate that EOSA-CNN achieved high validation accuracy. The single most striking observation to emerge from the comparison of the algorithms was that all the hybrid algorithms highly outperformed the traditional CNN and had a relative performance to EOSA-CNN.

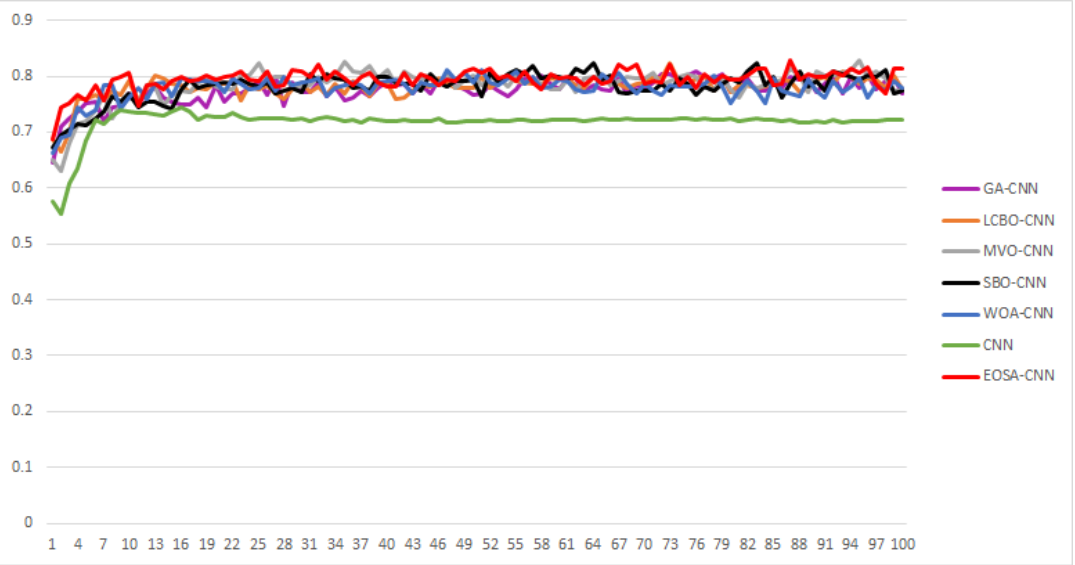


Figure 4.2: Explain the mean validation performance of EOSA-CNN compared to other metaheuristic algorithms and the traditional CNN.

4.3 Discussion and Comparison of Results

Lung cancer is an emerging cause of death worldwide due to the lack of adequate and effective diagnostic tools and proper treatment. In recent times, several state-of-the-art deep learning method-based optimization techniques have been employed to identify lung cancer in its early stage accurately. Deep learning techniques are extensively used in various domains of machine vision, including object detection, segmentation, and image classification. Although several studies have been obtained, none of these studies have been very effective in addressing some of the main challenges of lung cancer diagnosis at its early stages. In this study, we proposed a hybrid algorithm EOSA-CNN with selected preprocessing methods for lung cancer detection as an initial objective of the study was to identify. Thereafter, the proposed method's performance was compared to five hybrid models, namely GA-CNN, LCBO-CNN, MVO-CNN, SBO-CNN, WOA-CNN, and the traditional CNN, based on five experiments. We conducted all the experiments under the same machines and parameters configurations for a fair comparison. Furthermore, we used eight performance metrics in order to see the difference between the performance of the methods.

Also, for a fair comparison, we used 80% of the data selected randomly to train the models, while the 20% was left aside of the entire dataset, which served as a testing set for the final evaluation. In addition, the models were trained using 100 epochs and a batch size of 128 to ensure that the models were trained under similar conditions.

The first question in this study sought to determine a new CNN architecture that is hybridized with EOSA to improve lung cancer classification accuracy. The most interesting finding is that the proposed algorithm classifies the malignant labels with high accuracy (see the per-class performance in Table 4.7). The most prominent finding from the analysis is that the proposed algorithm outperforms the other state-of-the-art hybrid algorithms and the traditional CNN. Moreover, EOSA-CNN identified the negative cases with high accuracy equal to 90 %. This confirms that our proposed model successfully classifies positive and negative patients, contrary to what is reported for the traditional CNN.

In Table 4.8 below, the proposed EOSA-CNN hybrid algorithm performance is compared with those reported in similar studies. The classification accuracy obtained in the approach proposed in this study outperformed those seen in the works of Song et al. (2017), Zheng et al. (2021), Lu et al. (2021), Priyadharshini et al. (2021), and Wang et al. (2020). We see that the classification accuracy reported in the work of Priyadharshini et al. (2021), who also used a metaheuristic-based optimization approach, surpasses that reported in this study. However, the result of specificity and precision, which are 1.0 for both cases as obtained in the approach proposed in this study, confirms that classification accuracy alone is not sufficient to demonstrate the superiority of the methods. We noticed that the proposed method in this study gave an outstanding performance in its ability to eliminate the presence of false positives and ensure that our model correctly classifies negative cases as negative and positive cases as positive. Also, the value of 1.0 for specificity as reported for the method proposed in this study showed that the total number of actual negative cases (normal and benign) in our datasets discovered to be truly negative was very accurate. That means all negative cases were truly confirmed negative by our method. This is very important to rule out the possibility of false negative and false positive results. Yielding a zero-level for false positive and false negative rates, as seen by our proposed method, showed that the EOSA-CNN hybrid algorithm is not just good for classification accuracy but can be relied on for reliable results. This will boost confidence in the resulting output by the proposed algorithm when deployed for use. Therefore, this study has demonstrated the importance of using metaheuristic

algorithm to optimize CNN models hyperparameters to solve the difficult problem of selecting the best combination of weights and bias required for training a CNN model. Moreover, the approach also demonstrates that the combined methods can improve classification accuracy and general performance of classifying lung cancer in CT images.

Table 4.8: Performance comparison of the proposed method and some similar methods of CNN for classification of lung cancer

Author	Method	Dataset	Performance
Song et al. (2017)	CNN, DNN, and SAE	LIDC-IDRI	Accuracy of 84.15%, sensitivity of 83.96%, and specificity of 84.32%
Zheng et al. (2021)	Combination radiology analysis and malignancy evaluation network (R2MNet)	LIDC-IDRI	Area under curve (AUC) of 96.27%
Lu et al. (2021)	Marine predators metaheuristic algorithm and CNN	RIDER	93.4% accuracy, 98.4% sensitivity, and 97.1% specificity
Priyadharshini et al. (2021)	Bat-inspired Metaheuristic Convolutional Neural Network	LIDC-IDRI dataset	Accuracy of 97.43%
Wang et al. (2020)	RNN and medical-tomedical, transfer learning technique	Lung cancer CT scans	Accuracy of 85.71%
Our study	**EOSA-CNN and selected preprocessing methods**	**IQ-OTH/NCCD lung cancer**	**Accuracy of 93.21%, sensitivity of 0.9071, specificity of 1.0, precision of 1.0, F1-score of 0.9272, and Recall of 0.9071**

4.4 Study limitations

The major limitation of this study is the small sample size. Also, we did not consider solving the issue of imbalanced data approaches such as random under and over sampling, and cluster based over sampling, among others. It also proposed that future work should consider using more samples size as well as handling the class imbalance problem which might increase the models performances. Overall, the present results are significant compared to previous studies above. Hence, EOSA-CNN model outperforms the other hybrid algorithms and the traditional CNN in terms of seven metrics used. Furthermore, we need to see the EOSA performance in different medical problem.

Chapter 5

Conclusion and Future Research Direction

5.1 Conclusion

Lung cancer is one of the significant causes of death in the world. Also, lung cancer is characterized by the uncontrollable development of tumor cells that influence human survival. Several techniques have been introduced and presented using different data analysis approaches. Due to the increase of the CT images produced, visual interpretation may lead to a complex and error-prone process, which can cause delay in lung cancer diagnosis and detection. Therefore, early and precise detection of lung cancer is becoming a challenging task for physicians. Therefore, the necessity of lung cancer diagnosis and detection techniques is on the rise, and hence, hybrid optimization approaches based on robust CNN models for early detection of lung cancer are considered as best alternatives. Thus, reducing the mortality due to lung cancer and increasing human life quality.

In this study, We used CT scan images for lung cancer classification. We initially prepared the CT images using modern image processing techniques, including grayscale, Gaussian blur, segmentation, normalization, erosion and dilation, CLAHE, and wavelet transforms, to get the correct and instant results. These techniques help to enhance the image quality by removing unwanted noise and focusing on the tumor portion. Thereafter, we used the extracted features that might give a high and accurate diagnosis and detection of lung cancer. In addition, we designed a new CNN architecture by hybridizing classical CNN model with the EOSA metaheuristic algorithm. The performance of new hybrid EOSA-CNN model was compared with various CNN optimized metaheuristic algorithms, namely, GA, LCBO, MVO, SBO, and WOA. The prepared data is divided into 80% training set and 20% testing set. The significant

extracted features serve as input to the hybrid algorithms for the classification process.

The study also focused on resolving lung cancer classification problem by combining thr proposed CNN architecture with the nature-inspired EOSA metaheuristic. The EOSA optimization algorithm was specifically employed to optimize the solution vector of the traditional CNN architecture. So, the metaheuristic algorithm based on the propagation strategy of the Ebola virus was trained to investigate its ability to optimize the solutions in the search space. First, the CNN architecture was designed and then the solution vector was computed and extracted. The solution vector was passed on to the EOSA metaheuristic algorithm for optimization purposes. The outcome of the process was then supplied as the optimized set of weights and biases for fully training the CNN model. Considering the good performance seen during the training phase, we then used the trained model to test new samples for classification accuracy. A comparative analysis of the EOSA-CNN with other five hybrid algorithms and the traditional CNN was carried out and reported. The results revealed that EOSA-CNN achieved higher performance compared to the other metaheuristic algorithms and the traditional CNN in terms of accuracy, kappa, precision, recall, F1 score, specificity, and sensitivity metrics. Hence, the contribution of this study is the use of a good hybrid algorithm, a virus-based optimization technique to improve the solution vector of the proposed CNN architecture. Furthermore, the proposed model revealed well performance in classifying the malignant and benign lung cancer cases using CT scan images.

5.2 Future Research Direction

Further research could assess the effects of handling imbalanced data with hybrid algorithms. Moreover, more sample size would help establish greater accuracy on lung cancer classification. In addition, the robustness of the current model can further be tested and validated using other types of benchmarked datasets such as X-ray, MRI, and gene expression data. Moreover, the proposed technique and preprocessing methods can also be useful in the diagnoses different cancer types and subtypes. We suggest comparing the proposed EOSA-CNN metaheuristic algorithm with different CNN architectures such as GoogleNet, ZFNet, VGGNet, ResNet, and AlexNet. In future, the applied EOSA method may be considered for use in auto-designing deep learning models (e.g CNN) for classification of lung cancer problem. In addition, a hybrid of EOSA with other state-of-the-art metaheuristics algorithm needs to be considered for addressing the optimization task.

References

Abdullah, M. F., Mansor, M. S., Sulaiman, S. N., Osman, M. K., Marzuki, N. N. S. M., Isa, I. S., Karim, N. K. A., & Shuaib, I. L. (2019). A comparative study of image segmentation technique applied for lung cancer detection. In *2019 9th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, (pp. 72–77). IEEE.

Abugabah, A., AlZubi, A. A., Al-Obeidat, F., Alarifi, A., & Alwadain, A. (2020). Data mining techniques for analyzing healthcare conditions of urban space-person lung using meta-heuristic optimized neural networks. *Cluster Computing*, *23*, 1781–1794.

Ahmed, T., Parvin, M. S., Haque, M. R., Uddin, M. S., et al. (2020). Lung cancer detection using ct image based on 3d convolutional neural network. *Journal of Computer and Communications*, *8*(03), 35.

Akay, B., Karaboga, D., & Akay, R. (2021). A comprehensive survey on optimizing deep learning models by metaheuristics. *Artificial Intelligence Review*, (pp. 1–66).

Al-Huseiny, M. S., & Sajit, A. S. (2021). Transfer learning with googlenet for detection of lung cancer. *Indonesian Journal of Electrical Engineering and Computer Science*, *22*(2), 1078–1086.

Al-Yasriy, H. F., AL-Husieny, M. S., Mohsen, F. Y., Khalil, E. A., & Hassan, Z. S. (2020). Diagnosis of lung cancer based on ct scans using cnn. In *IOP Conference Series: Materials Science and Engineering*, vol. 928, (p. 022035). IOP Publishing.

Alagarsamy, S., Subramanian, R. R., Shree, T., Balasubramanian, M., Govindaraj, V., et al. (2021). Prediction of lung cancer using meta-heuristic based optimization technique: Crow search technique. In *2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, (pp. 186–191). IEEE.

Ali, S. M., Mohamed, A.-A. A., & Hemeida, A. (2019). Impact of optimal allocation of renewable distributed generation in radial distribution systems based on different

optimization algorithms. In *2019 International Conference on Innovative Trends in Computer Engineering (ITCE)*, (pp. 472–478). IEEE.

Almomani, O. (2021). A hybrid model using bio-inspired metaheuristic algorithms for network intrusion detection system. *CMC-COMPUTERS MATERIALS & CONTINUA*, *68*(1), 409–429.

Alrahhal, M. S., & Alqhtani, E. (2021). Deep learning-based system for detection of lung cancer using fusion of features.

Balan, S. (????). Metaheuristics in optimization: Algorithmic perspective.

Bandaru, S., & Deb, K. (2016). Metaheuristic techniques. *Decision sciences*, (pp. 693–750).

Bengio, Y. (2009). *Learning deep architectures for AI*. Now Publishers Inc.

Benmessahel, I., Xie, K., & Chellal, M. (2020). A new competitive multiverse optimization technique for solving single-objective and multiobjective problems. *Engineering Reports*, *2*(3), e12124.

Berman, J. J. (2004). Tumor classification: molecular analysis meets aristotle. *BMC cancer*, *4*(1), 1–9.

Bhandary, A., Prabhu, G. A., Rajinikanth, V., Thanaraj, K. P., Satapathy, S. C., Robbins, D. E., Shasky, C., Zhang, Y.-D., Tavares, J. M. R., & Raja, N. S. M. (2020). Deep-learning framework to detect lung abnormality–a study with chest x-ray and lung ct scan images. *Pattern Recognition Letters*, *129*, 271–278.

Bhatia, S., Sinha, Y., & Goel, L. (2019). Lung cancer detection: a deep learning approach. In *Soft Computing for Problem Solving*, (pp. 699–705). Springer.

Bigna, J. J., & Noubiap, J. J. (2019). The rising burden of non-communicable diseases in sub-saharan africa. *The Lancet Global Health*, *7*(10), e1295–e1296.

Boix, M., & Cantó, B. (2010). Wavelet transform application to the compression of images. *Mathematical and computer modelling*, *52*(7-8), 1265–1270.

Caprara, G. (2021). Mediterranean-type dietary pattern and physical activity: The winning combination to counteract the rising burden of non-communicable diseases (ncds). *Nutrients*, *13*(2), 429.

Cengil, E., & Cinar, A. (2018). A deep learning based approach to lung cancer identification. In *2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*, (pp. 1–5). IEEE.

Chadha, P., & Singh, G. (2012). Classification rules and genetic algorithm in data mining. *Global Journal of Computer Science and Technology Software & Data Engineering*, *12*(15), 50–54.

Chaturvedi, P., Jhamb, A., Vanani, M., & Nemade, V. (2021). Prediction and classification of lung cancer using machine learning techniques. In *IOP Conference Series: Materials Science and Engineering*, vol. 1099, (p. 012059). IOP Publishing.

Chen, J., Zeng, H., Zhang, C., Shi, Z., Dekker, A., Wee, L., & Bermejo, I. (2021). Lung cancer diagnosis using deep attention based on multiple instance learning and radiomics. *arXiv preprint arXiv:2104.14655*.

Chon, A., Balachandar, N., & Lu, P. (2017). Deep convolutional neural networks for lung cancer detection. *Standford University*.

Ciregan, D., Meier, U., & Schmidhuber, J. (2012). Multi-column deep neural networks for image classification. In *2012 IEEE conference on computer vision and pattern recognition*, (pp. 3642–3649). IEEE.

Coudray, N., Ocampo, P. S., Sakellaropoulos, T., Narula, N., Snuderl, M., Fenyö, D., Moreira, A. L., Razavian, N., & Tsirigos, A. (2018). Classification and mutation prediction from non–small cell lung cancer histopathology images using deep learning. *Nature medicine*, *24*(10), 1559–1567.

Daneshjou, R., He, B., Ouyang, D., & Zou, J. (2021). How to evaluate deep learning for cancer diagnostics–factors and recommendations. *Biochimica et Biophysica Acta (BBA)-Reviews on Cancer*, (p. 188515).

Das, S., Mishra, S., & Senapati, M. R. (2020). New approaches in metaheuristic to classify medical data using artificial neural network. *Arabian Journal for Science and Engineering*, *45*(4), 2459–2471.

Davies, P. C. (1978). Thermodynamics of black holes. *Reports on Progress in Physics*, *41*(8), 1313.

Depeursinge, A., Fageot, J., & Al-Kadi, O. (2017). Fundamentals of texture processing for biomedical image analysis. Tech. rep., Citeseer.

Elbashir, M. K., Ezz, M., Mohammed, M., & Saloum, S. S. (2019). Lightweight convolutional neural network for breast cancer classification using rna-seq gene expression data. *IEEE Access*, *7*, 185338–185348.

Globocan (2021). Africa source: Globocan 2020. Tech. rep., The Global Cancer Observatory.

Hamdi, Y., Abdeljaoued-Tej, I., Zatchi, A. A., Abdelhak, S., Boubaker, S., Brown, J. S., & Benkahla, A. (2021). Cancer in africa: The untold story. *Frontiers in oncology*, *11*.

Hassen, D. B., & Zakour, S. B. (2019). The performance of novel wavelet for lung image analysis. *Clinical Medical Reviews and Case Reports*, *6*(7), 1–5.

Hatuwal, B. K., & Thapa, H. C. (2020). Lung cancer detection using convolutional neural network on histopathological images. *Int. J. Comput. Trends Technol*, *68*, 21–24.

Hemeida, M. G., Alkhalaf, S., Senjyu, T., Ibrahim, A., Ahmed, M., & Bahaa-Eldin, A. M. (2021). Optimal probabilistic location of dgs using monte carlo simulation based different bio-inspired algorithms. *Ain Shams Engineering Journal*.

Ibrahim, N. M., Abou ElFarag, A., & Kadry, R. (2021). Gaussian blur through parallel computing. In *IMPROVE*, (pp. 175–179).

Irawanto, M. A., Setianingsih, C., & Irawan, B. (2022). Detection of traffic density with image processing using pin hole algorithm. *IIUM Engineering Journal*, *23*(1), 244–257.

Jamil, M., & Yang, X.-S. (2013). A literature survey of benchmark functions for global optimisation problems. *International Journal of Mathematical Modelling and Numerical Optimisation*, *4*(2), 150–194.

Jianhao, W., Long, W., Lijie, C., & Tian, G. (2021). Enhanced whale optimization algorithm for large-scale global optimization problems. In *2021 International Conference on Computer Communication and Artificial Intelligence (CCAI)*, (pp. 180–187). IEEE.

Kalaivani, N., Manimaran, N., Sophia, S., & Devi, D. (2020). Deep learning based lung cancer detection and classification. In *IOP Conference Series: Materials Science and Engineering*, vol. 994, (p. 012026). IOP Publishing.

Kareem, H. F., AL-Husieny, M. S., Mohsen, F. Y., Khalil, E. A., & Hassan, Z. S. (2021). Evaluation of svm performance in the detection of lung cancer in marked ct scan dataset. *Indonesian Journal of Electrical Engineering and Computer Science*, *21*(3), 1731–1738.

Kaur, S. (2015). Noise types and various removal techniques. *International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)*, *4*(2), 226–230.

Kerneler, K. (????). The Iraq-Oncology Teaching Hospital/ National Center for Cancer Diseases (IQ-OTH/NCCD) lung cancer dataset, year = 2020, url = https://www.kaggle.com/kerneler/starter-the-iq-oth-nccd-lung-cancer-09c3a8c9-4/data, urldate = 2021-09-15.

Khamparia, A., Singh, A., Anand, D., Gupta, D., Khanna, A., Arun Kumar, N., & Tan, J. (2020). A novel deep learning-based multi-model ensemble method for the prediction of neuromuscular disorders. *Neural computing and applications*, *32*(15), 11083–11095.

Khatami, A., Khosravi, A., Nguyen, T., Lim, C. P., & Nahavandi, S. (2017). Medical image analysis using wavelet transform and deep belief networks. *Expert Systems with Applications*, *86*, 190–198.

Khatri, A., Gaba, A., Rana, K., & Kumar, V. (2020). A novel life choice-based optimizer. *Soft Computing*, *24*(12), 9121–9141.

Khoury, J., Ovrut, B. A., Seiberg, N., Steinhardt, P. J., & Turok, N. (2002). From big crunch to big bang. *Physical Review D*, *65*(8), 086007.

Khumancha, M. B., Barai, A., & Rao, C. R. (2019). Lung cancer detection from computed tomography (ct) scans using convolutional neural network. In *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, (pp. 1–7). IEEE.

Kirienko, M., Sollini, M., Silvestri, G., Mognetti, S., Voulaz, E., Antunovic, L., Rossi, A., Antiga, L., & Chiti, A. (2018). Convolutional neural networks promising in lung cancer t-parameter assessment on baseline fdg-pet/ct. *Contrast media & molecular imaging*, *2018*.

Kornilov, A. S., & Safonov, I. V. (2018). An overview of watershed algorithm implementations in open source libraries. *Journal of Imaging*, *4*(10), 123.

Kriegsmann, M., Haag, C., Weis, C.-A., Steinbuss, G., Warth, A., Zgorzelski, C., Muley, T., Winter, H., Eichhorn, M. E., Eichhorn, F., et al. (2020). Deep learning for the classification of small-cell and non-small-cell lung cancer. *Cancers*, *12*(6), 1604.

Kuruvilla, J., & Gunavathi, K. (2014). Lung cancer classification using neural networks for ct images. *Computer methods and programs in biomedicine*, *113*(1), 202–209.

Li, J., Fong, S., Liu, L.-s., Dey, N., Ashour, A. S., & Moraru, L. (2019). Dual feature selection and rebalancing strategy using metaheuristic optimization algorithms in x-ray image datasets. *Multimedia Tools and Applications*, *78*(15), 20913–20933.

Li, J., Lam, A. S., Yau, S. T., Yiu, K. K., & Tsoi, K. K. (2021). Antihypertensive treatments and risks of lung cancer: a large population-based cohort study in hong kong. *BMC cancer*, *21*(1), 1–9.

Li, L., & Weinberg, C. R. (2003). Gene selection and sample classification using a genetic algorithm and k-nearest neighbor method. In *A Practical Approach to Microarray Data Analysis*, (pp. 216–229). Springer.

Lu, X., Nanehkaran, Y., & Karimi Fard, M. (2021). A method for optimal detection of lung cancer based on deep learning optimized by marine predators algorithm. *Computational Intelligence and Neuroscience*, *2021*.

Makaju, S., Prasad, P., Alsadoon, A., Singh, A., & Elchouemi, A. (2018). Lung cancer detection using ct scan images. *Procedia Computer Science*, *125*, 107–114.

Maleki, N., Zeinali, Y., & Niaki, S. T. A. (2021). A k-nn method for lung cancer prognosis with the use of a genetic algorithm for feature selection. *Expert Systems with Applications*, *164*, 113981.

Mapanga, W., Norris, S. A., Chen, W. C., Blanchard, C., Graham, A., Baldwin-Ragaven, L., Boyles, T., Donde, B., Greef, L., Huddle, K., et al. (2021). Consensus study on the health system and patient-related barriers for lung cancer management in south africa. *Plos one*, *16*(2), e0246716.

Masud, M., Sikder, N., Nahid, A.-A., Bairagi, A. K., & AlZain, M. A. (2021). A machine learning approach to diagnosing lung and colon cancer using a deep learning-based classification framework. *Sensors*, *21*(3), 748.

Miah, M. B. A., & Yousuf, M. A. (2015). Detection of lung cancer from ct image using image processing and neural network. In *2015 International conference on electrical engineering and information communication technology (ICEEICT)*, (pp. 1–6). ieee.

Mirjalili, S., & Lewis, A. (2016). The whale optimization algorithm. *Advances in engineering software*, *95*, 51–67.

Mirjalili, S., Mirjalili, S. M., & Hatamlou, A. (2016). Multi-verse optimizer: a nature-inspired algorithm for global optimization. *Neural Computing and Applications*, *27*(2), 495–513.

Mohammed, M., Mwambi, H., Mboya, I. B., Elbashir, M. K., & Omolo, B. (2021). A stacking ensemble deep learning approach to cancer type classification based on tcga data. *Scientific reports*, *11*(1), 1–22.

Moosavi, S. H. S., & Bardsiri, V. K. (2017). Satin bowerbird optimizer: A new optimization algorithm to optimize anfis for software development effort estimation. *Engineering Applications of Artificial Intelligence*, *60*, 1–15.

Motohiro, A., Ueda, H., Komatsu, H., Yanai, N., Mori, T., for Lung Cancer, N. C. H. S. G., et al. (2002). Prognosis of non-surgically treated, clinical stage i lung cancer patients in japan. *Lung Cancer*, *36*(1), 65–69.

Nanglia, P., Kumar, S., Mahajan, A. N., Singh, P., & Rathee, D. (2021). A hybrid algorithm for lung cancer classification using svm and neural networks. *ICT Express*, *7*(3), 335–341.

Neal Joshua, E. S., Bhattacharyya, D., Chakkravarthy, M., & Byun, Y.-C. (2021). 3d cnn with visual insights for early detection of lung cancer using gradient-weighted class activation. *Journal of Healthcare Engineering*, *2021*.

Ning, G.-Y., & Cao, D.-Q. (2021). Improved whale optimization algorithm for solving constrained optimization problems. *Discrete Dynamics in Nature and Society*, *2021*.

Olsen, M. (2015). *Cancer in Sub-Saharan Africa: The need for new paradigms in global health*. Boston University Frederick S. Pardee Center for the Study of the Longer ....

Organization, W. H. (2002). *National cancer control programmes: policies and managerial guidelines*. World Health Organization.

Oyelade, O. N., & Ezugwu, A. E. (2021a). A bioinspired neural architecture search based convolutional neural network for breast cancer detection using histopathology images. *Scientific Reports*, *11*(1), 1–28.

Oyelade, O. N., & Ezugwu, A. E. (2021b). Characterization of abnormalities in breast cancer images using nature-inspired metaheuristic optimized convolutional neural networks model. *Concurrency and Computation: Practice and Experience*, (p. e6629).

Oyelade, O. N., & Ezugwu, A. E. (2021c). Ebola optimization search algorithm (eosa): A new metaheuristic algorithm based on the propagation model of ebola virus disease. *arXiv preprint arXiv:2106.01416*.

Park, H., & Monahan, C. (2019). Genetic deep learning for lung cancer screening. *arXiv preprint arXiv:1907.11849*.

Pereira, T., Freitas, C., Costa, J. L., Morgado, J., Silva, F., Negrão, E., de Lima, B. F., da Silva, M. C., Madureira, A. J., Ramos, I., et al. (2021). Comprehensive perspective for lung cancer characterisation based on ai solutions using ct images. *Journal of Clinical Medicine*, *10*(1), 118.

Prasad, P., Prasad, D., & Rao, G. S. (2016). Performance analysis of orthogonal and biorthogonal wavelets for edge detection of x-ray images. *Procedia Computer Science*, *87*, 116–121.

Priyadharshini, P., Zoraida, B., et al. (2021). Bat-inspired metaheuristic convolutional neural network algorithms for cad-based lung cancer prediction. *Journal of Applied Science and Engineering*, *24*(1), 65–71.

Rajakumari, R., & Kalaivani, L. (2021). Breast cancer detection and classification using deeper convolutional neural networks based on wavelet packet decomposition techniques.

Rajasenbagam, T., & Jeyanthi, S. (2020). Image enhancement filtering techniques to enhance ct images of lung cancer. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, *9*(4), 241–248.

Rere, L., Fanany, M. I., & Arymurthy, A. M. (2016). Metaheuristic algorithms for convolution neural network. *Computational intelligence and neuroscience*, *2016*.

Revathi, M., Jeya, I. J. S., & Deepa, S. (2020). Deep learning-based soft computing model for image classification application. *Soft Computing*, *24*, 18411–18430.

Rosales-Muñoz, A. A., Grisales-Noreña, L. F., Montano, J., Montoya, O. D., & Perea-Moreno, A.-J. (2021). Application of the multiverse optimization method to solve the optimal power flow problem in direct current electrical networks. *Sustainability*, *13*(16), 8703.

Sajja, T., Devarapalli, R., & Kalluri, H. (2019). Lung cancer detection based on ct scan images by using deep transfer learning. *Traitement du Signal*, *36*(4), 339–344.

Senthil Kumar, K., Venkatalakshmi, K., & Karthikeyan, K. (2019). Lung cancer detection using image segmentation by means of various evolutionary algorithms. *Computational and mathematical methods in medicine*, *2019*.

Shakeel, P. M., Burhanuddin, M. A., & Desa, M. I. (2019). Lung cancer detection from ct image using improved profuse clustering and deep learning instantaneously trained neural networks. *Measurement*, *145*, 702–712.

Shan, R., & Rezaei, T. (2021). Lung cancer diagnosis based on an ann optimized by improved teo algorithm. *Computational Intelligence and Neuroscience*, *2021*.

Shankar, A., Saini, D., Dubey, A., Roy, S., Bharati, S. J., Singh, N., Khanna, M., Prasad, C. P., Singh, M., Kumar, S., et al. (2019). Feasibility of lung cancer screening in developing countries: challenges, opportunities and way forward. *Translational lung cancer research*, *8*(Suppl 1), S106.

Shoaib, M., Naseem, R., & Dar, A. H. (2013). Automated segmentation of lungs in computed tomographic images. *Eur J Sci Res*, *98*, 45–54.

Siddiqi, M. H., Ahmad, I., & Sulaiman, S. B. (2009). Weed recognition based on erosion and dilation segmentation algorithm. In *2009 International Conference on Education Technology and Computer*, (pp. 224–228). IEEE.

Siegel, R. L., Miller, K. D., Fuchs, H. E., & Jemal, A. (2021). Cancer statistics, 2021. *CA: a cancer journal for clinicians*, *71*(1), 7–33.

Singh, G. A. P., & Gupta, P. (2019). Performance analysis of various machine learning-based approaches for detection and classification of lung cancer in humans. *Neural Computing and Applications*, *31*(10), 6863–6877.

Soille, P. (2013). *Morphological image analysis: principles and applications*. Springer Science & Business Media.

Song, Q., Zhao, L., Luo, X., & Dou, X. (2017). Using deep learning for classification of lung nodules on computed tomography images. *Journal of healthcare engineering*, *2017*.

Soydaner, D. (2020). A comparison of optimization algorithms for deep learning. *International Journal of Pattern Recognition and Artificial Intelligence*, *34*(13), 2052013.

Sridhar, S., Kumar, P. R., & Ramanaiah, K. (2014). Wavelet transform techniques for image compression-an evaluation. *International journal of image, graphics and signal processing*, *6*(2), 54.

Sujitha, R., & Seenivasagam, V. (2021). Classification of lung cancer stages with machine learning over big data healthcare framework. *Journal of Ambient Intelligence and Humanized Computing*, *12*, 5639–5649.

Sun, R.-Y. (2020). Optimization for deep learning: An overview. *Journal of the Operations Research Society of China*, *8*(2), 249–294.

Sun, W., Zheng, B., & Qian, W. (2017). Automatic feature learning using multichannel roi based on deep structured algorithms for computerized lung cancer diagnosis. *Computers in biology and medicine*, *89*, 530–539.

Sung, H., Ferlay, J., Siegel, R. L., Laversanne, M., Soerjomataram, I., Jemal, A., & Bray, F. (2021). Global cancer statistics 2020: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: a cancer journal for clinicians*, *71*(3), 209–249.

Tambe, S. B., Kulhare, D., Nirmal, M., & Prajapati, G. (2013). Image processing (ip) through erosion and dilation methods.

Tegmark, M. (2003). Parallel universes. *Scientific American*, *288*(5), 40–51.

Toğaçar, M., Ergen, B., & Cömert, Z. (2020). Detection of lung cancer on chest ct images using minimum redundancy maximum relevance feature selection method with convolutional neural networks. *Biocybernetics and Biomedical Engineering*, *40*(1), 23–39.

Wang, S., Dong, L., Wang, X., & Wang, X. (2020). Classification of pathological types of lung cancer from ct images by deep residual neural networks with transfer learning strategy. *Open Medicine*, *15*(1), 190–197.

Wang, Y., & Wang, J. (2020). Modelling and prediction of global non-communicable diseases. *BMC Public Health*, *20*, 1–13.

Woodman, C., Vundu, G., George, A., & Wilson, C. M. (2021). Applications and strategies in nanodiagnosis and nanotherapy in lung cancer. In *Seminars in cancer biology*, vol. 69, (pp. 349–364). Elsevier.

Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. *Insights into imaging*, *9*(4), 611–629.

Yamunadevi, M., & Ranjani, S. S. (2021). Efficient segmentation of the lung carcinoma by adaptive fuzzy–glcm (af-glcm) with deep learning based classification. *Journal of Ambient Intelligence and Humanized Computing*, *12*(5), 4715–4725.

Yang, X.-S. (2011). Metaheuristic optimization: algorithm analysis and open problems. In *International Symposium on Experimental Algorithms*, (pp. 21–32). Springer.

Zhang, J., Pang, H., Cai, W., & Yan, Z. (2021a). Using image processing technology to create a novel fry counting algorithm. *Aquaculture and Fisheries*.

Zhang, S., Zhou, Y., & Luo, Q. (2021b). A complex-valued encoding satin bowerbird optimization algorithm for global optimization. *Evolving Systems*, *12*(1), 191–205.

Zheng, S., Shen, Z., Peia, C., Ding, W., Lin, H., Zheng, J., Pan, L., Zheng, B., & Huang, L. (2021). Interpretative computer-aided lung cancer diagnosis: from radiology analysis to malignancy evaluation. *arXiv preprint arXiv:2102.10919*.