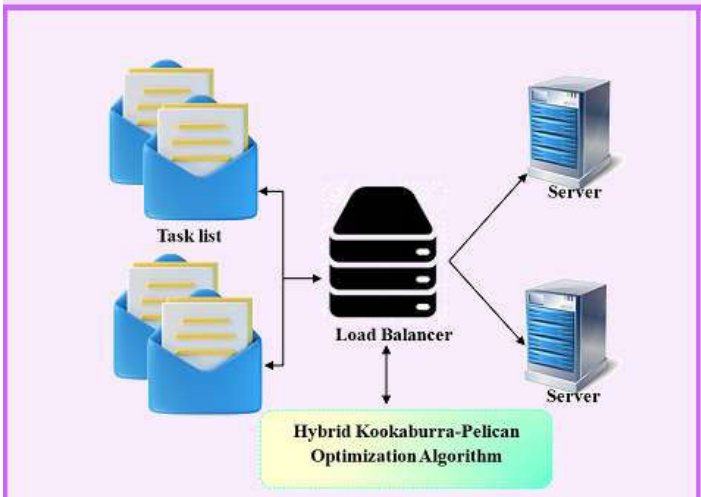
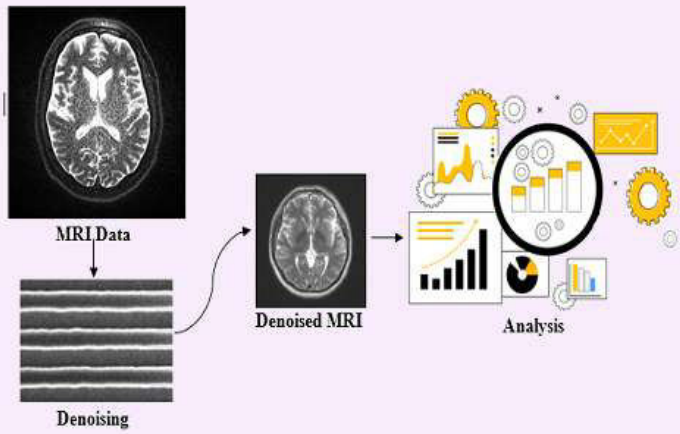
Volume 02, Number 04, July - August (2024)

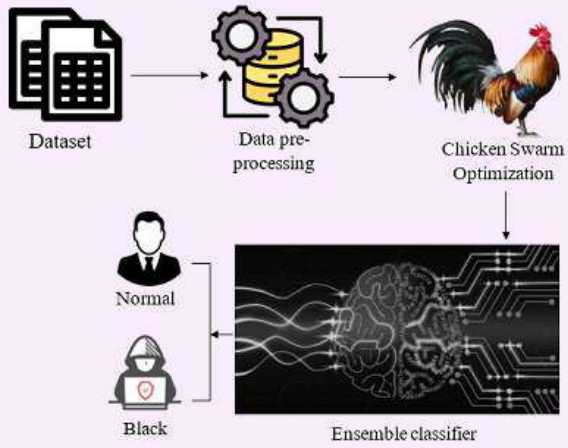
IJDSAI

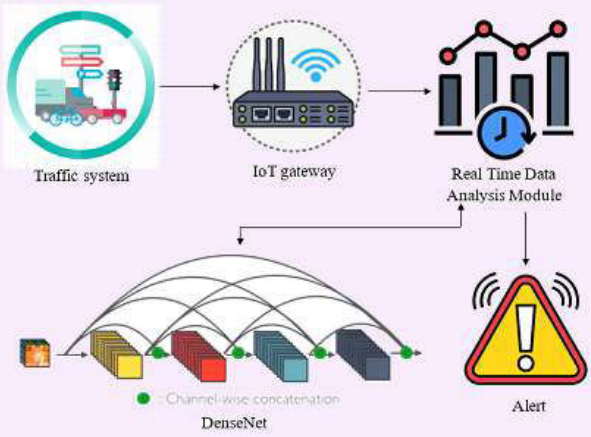
ISSN: 2584-1041

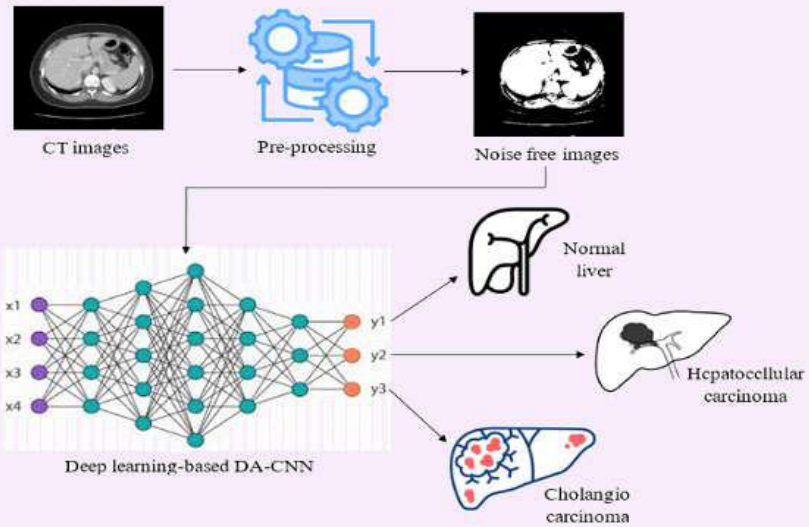
International Journal of Data Science and Artificial Intelligence











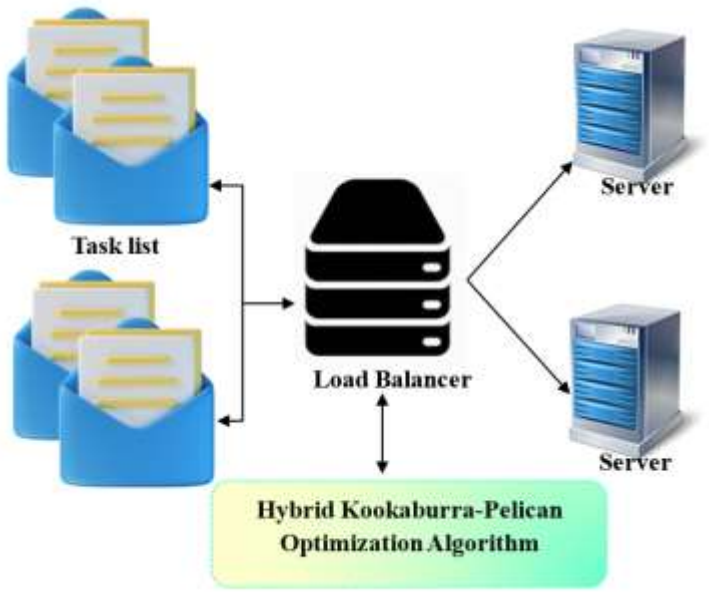
International Journal of Data Science and Artificial Intelligence

IJDSAI

1. DYNAMIC LOAD BALANCING IN CLOUD COMPUTING USING HYBRID KOOKABURRA-PELICAN OPTIMIZATION ALGORITHMS

G. Saranya, G. Belshia Jebamalar and Chukka Santhaiah

Abstract – Cloud Computing (CC) technology facilitates virtualized computer resources to users via service providers. Load balancing assumes a critical role in distributing dynamic workloads across cloud systems, ensuring equitable resource allocation without overwhelming or underutilizing virtual machines (VMs). However, uneven workload distribution poses a significant challenge in cloud data centers, hindering efficient resource utilization. To address these issues, this paper proposes a novel Dynamic Efficient Load Balancing in Cloud using kookaburra Infused pelican Optimization for virtUal Server (DELICIOUS) is developed for effective load balancing process in cloud computing environment. The Hybrid Kookaburra-Pelican Optimization Algorithm (HK-POA) is implemented for offloading decisions which optimizes resource allocation and enhances user experiences. The evaluation of the performance of the DELICIOUS framework involves a thorough assessment that includes essential metrics such as throughput, execution time, latency, waiting time, computational complexity, and computational cost. The simulation experiments of the proposed DELICIOUS framework are conducted using CloudSim and achieves a better throughput of 1206.6 Kbps whereas, the GRAF, QoDA-LB, and RATS-HM technique attains 865 Kbps, 943.4 Kbps, and 984.6 Kbps respectively for intelligent load balancing in cloud networks.

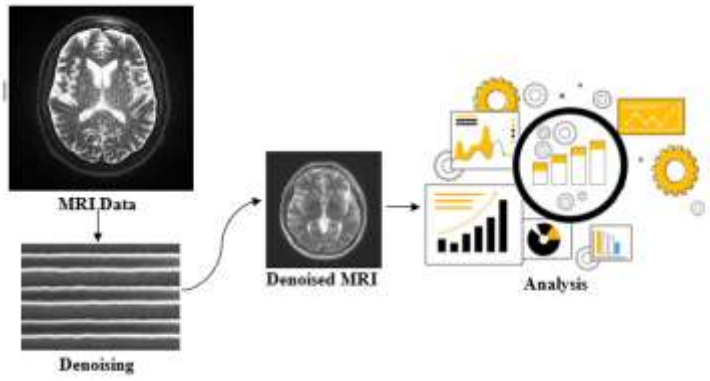


Keywords – Load Balancing, Cloud Computing, Kookaburra Optimization Algorithm, Pelican Optimization Algorithm

2. IN-DEPTH EXPLORATION AND COMPARATIVE ASSESSMENT OF CUTTING-EDGE ALGORITHMS FOR IMPULSE NOISE ATTENUATION IN CORRUPTED VISUAL DATA

S. Prathiba and B. Sivagami

Abstract – Image denoising is a vital process in image pre-processing, particularly for applications focused on image-based objectives. This process, which occurs during image acquisition and transmission, is crucial for enhancing image quality to facilitate subsequent analysis by medical image processing algorithms. Given its importance in improving medical images, image denoising has become a prominent research focus. This paper explores the latest advancements in denoising techniques specifically tailored for magnetic resonance imaging (MRI), offering a detailed examination of their publication details, underlying methodologies, strengths, limitations, and accuracy metrics, including peak signal-to-noise ratio (PSNR). The study provides a thorough review of contemporary denoising strategies proposed by researchers such as Taherkhani et al., Zhang et al., Yuan et al., and Chen et al., among others, presenting an in-depth survey of current denoising algorithms. Understanding these methods is critical for selecting robust denoising techniques capable of mitigating artifacts like salt-and-pepper noise, which is essential for effective medical image segmentation. This paper aims to provide valuable insights into denoising methodologies, thereby advancing MRI image processing in the medical domain.

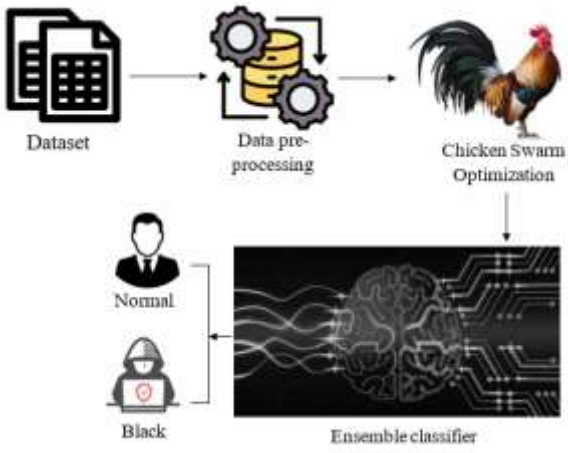


Keywords –Image Denoising, Magnetic Resonance Imaging (MRI), Medical Image Processing, Peak Signal-to-Noise Ratio (PSNR), Noise Reduction Algorithms.

3. CHICKEN SWARM OPTIMIZATION BASED ENSEMBLED LEARNING CLASSIFIER FOR BLACK HOLE ATTACK IN WIRELESS SENSOR NETWORK

K. Vijayan, S.V. Harish and R.A. Mabel Rose

Abstract – Wireless Sensor Networks (WSNs) are an inevitable technology prevalently used in various critical and remote monitoring applications. The security of WSNs is compromised by various attacks in wireless medium. Even though, various attacks are present, the Black hole attack degrades the network performance and resource utilization resulting in poor network lifetime. Therefore, the proposed research suggests an effective Intrusion Detection System for WSN to detect and classify black hole attacks based on ensemble ML classifiers. The BDD dataset is used for the analysis which is subjected to Chicken Swarm Optimization based feature selection. The selected features are balanced through SMOTE and TOMEK based STL data balancing module. An ensemble of five baseline ML classifiers such as SMO, NB, J48, KNN and RF utilizing voting ensemble approach is suggested to classify the attacks in the dataset. The performance of the algorithm is analyzed through evaluation metrics such as accuracy, precision, recall and F1-score. The comparison of proposed model with six ML and DL classifiers exposes the superiority of the proposed model's classification performance.

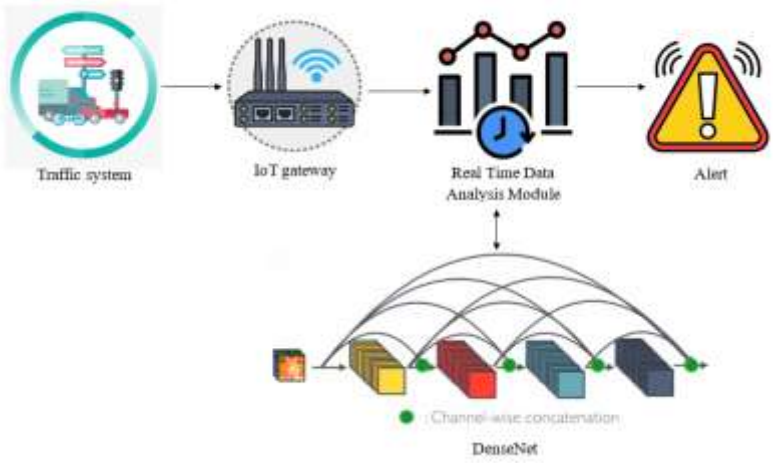


Keywords – WSN, Black hole attack, SMOTE, BDD dataset, Chicken Swarm Optimization, Ensemble classifier.

4. IOT BASED AIR QUALITY MONITORING USING DENSENET IN URBAN AREAS

M. Devaki, Jeyaraman Sathiamoorthy and M. Usha

Abstract – Internet of Things is being used more and more in the control and monitoring of air quality. Real-time data regarding air pollutants and other environmental parameters can be gathered by deploying IoT devices with sensors and connectivity capabilities. Rapid urbanization and industry cause increasingly serious problems with air quality. A significant challenge in the current air quality monitoring system is its limited spatial coverage and accuracy. In this paper, a novel air quality monitoring using IoT is proposed to monitor the quality of the air efficiently in real time. Sensors are placed in the various traffic system to collect environmental data and processed it in Real Time Data Analytics Module (RTDM). DenseNet is used to predict the quality of air and classified into three classes namely pure, impure, and normal. The efficacy of the proposed technique has been evaluated using assessment actions such as accuracy, time efficiency, precision, F1 score, RMSE, MAPE, and MAE. By the comparison analysis, the proposed technique's accuracy rate is 10.08%, 17.64%, and 34.34% higher than the existing Ide Air, SMOTEDNN, and ETAPM-AIT techniques respectively.

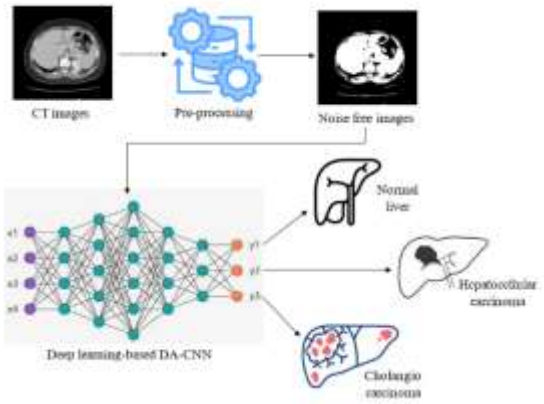


Keywords – Air pollution, DenseNet, Sensors, Internet of Things, Real-Time Data Analytics Module.

5. CLASSIFICATION OF LIVER CANCER VIA DEEP LEARNING BASED DILATED ATTENTION CONVOLUTIONAL NEURAL NETWORK

R. Ramani, K. Vimala Devi, P. Thiruselvan and M. Umamaheswari

Abstract – Liver cancer occur when normal cells develop aberrant DNA alterations and reproduce uncontrollably. Patients with cirrhosis, hepatitis B or C, or both have an increased risk of developing the progressing stage of cancer. The radiologists spend more time for detecting the liver cancer when analysing with traditional methods. Early detection of liver cancer can help doctors and radiation therapists identify the tumours. However, manual identification of liver cancer is time-intensive and challenging process in the current scenario. In this work, an automated deep learning network is designed to classify the liver cancer in its initial phase. At first, the CT scans are gathered from the publicly available LiTS database and these gathered images are pre-processed using Gaussian filter is used for reducing the noises and to smoothen the edges. The liver region is segmented using Enhanced otsu (EM) method is utilized to segment the liver region separately from the pre-processed input images. Afterwards, Dilated Convolutional Neural Network (DCNN) with the attention block is employed for classifying the liver cancer into tri-classes such as normal controls (NC), hepatocellular carcinoma (HCC) and cholangiocarcinoma (CC) cases based on the extracted features. The efficiency of the proposed DA-CNN is evaluated using the attributes viz., accuracy, sensitivity, precision, specificity, and F1-score values are computed as classification results. The experimental fallouts disclose that the DA-CNN attains an accuracy range of 98.20%. Moreover, the proposed DA-CNN advances the overall accuracy by 3.25%, 5.29%, and 0.99% better than Optimised GAN, OPBS-SSHC, HFCNN respectively.



Keywords – Liver cancer, Deep learning, CT images, Attention block, Enhanced otsu method.

International Journal of Data Science and Artificial Intelligence (IJDSAI)
Volume 02, Issue 04, July - August (2024)

RESEARCH ARTICLE

DYNAMIC LOAD BALANCING IN CLOUD COMPUTING USING HYBRID KOOKABURRA-PELICAN OPTIMIZATION ALGORITHMS

G. Saranya [1, *], G. Belshia Jebamalar [2] and Chukka Santhaiah [3]

[1] Department of Computer Science and Engineering, S.A. Engineering College, Poonamallee, Thiruverkadu, Tamil Nadu 600077 India.
[2] Department of Computer Science and Engineering, SA Engineering College, Poonamallee, Thiruverkadu, Tamil Nadu 600077 India.
[3] Department of Computer Science and Engineering, Sri Venkateswara College of Engineering, Tripati, Tamil Nadu India.

[*]*Corresponding e-mail: saran03ganesan@gmail.com*

Abstract – Cloud Computing (CC) technology facilitates virtualized computer resources to users via service providers. Load balancing assumes a critical role in distributing dynamic workloads across cloud systems, ensuring equitable resource allocation without overwhelming or underutilizing virtual machines (VMs). However, uneven workload distribution poses a significant challenge in cloud data centers, hindering efficient resource utilization. To address these issues, this paper proposes a novel Dynamic Efficient Load Balancing in Cloud using kookaburra Infused pelican Optimization for virtUal Server (DELICIOUS) is developed for effective load balancing process in cloud computing environment. The Hybrid Kookaburra-Pelican Optimization Algorithm (HK-POA) is implemented for offloading decisions which optimizes resource allocation and enhances user experiences. The evaluation of the performance of the DELICIOUS framework involves a thorough assessment that includes essential metrics such as throughput, execution time, latency, waiting time, computational complexity, and computational cost. The simulation experiments of the proposed DELICIOUS framework are conducted using CloudSim and achieves a better throughput of 1206.6 Kbps whereas, the GRAF, QoDA-LB, and RATS-HM technique attains 865 Kbps, 943.4 Kbps, and 984.6 Kbps respectively for intelligent load balancing in cloud networks.

Keywords *– Load Balancing, Cloud Computing, Kookaburra Optimization Algorithm, Pelican Optimization Algorithm.*

1. INTRODUCTION

Cloud Computing enables the efficient utilization of computing resources through its dynamic service model, requiring adaptive resource allocation and scalability to ensure Quality-of-Service (QoS) while minimizing resource usage [1, 2]. These resources cover computing power, storage, databases, networking, planning, resource finding, security, and privacy. Load balancing involves distributing workload effectively across multiple computing platforms, aiming to optimize system output, resource utilization, and VM performance metrics. Various load-balancing algorithms are employed by the cloud system to achieve resource efficiency [3, 4].

Load balancing algorithms varies depending on the system's condition which distinguishes static and dynamic methods. Static algorithms rely on heuristics and are contingent on the current system state, while dynamic algorithms utilize metaheuristics and operate independently of specific conditions [5, 6]. Dynamic algorithms are particularly effective in environments where the volume of requests and VMs varies significantly. These algorithms perform better in redistributing workloads among VMs to rectify load imbalances [7, 8].

Load refers to the tasks allocated to VMs, which may lead to imbalances due to underutilization or overutilization. Overutilization happens when tasks allocated to a VM exceed its capacity threshold, whereas underutilization occurs when a VM can handle more tasks than it currently hosts [9-10]. Therefore, load balancing is crucial for maintaining equilibrium among cloud resources. Load balancing and task scheduling are both required to accomplish this balance and ensure equitable allocation among virtual machines [11].

The load balancing and task scheduling is essential for meeting QoS requirements which falls into the category of NP-hard problems due to the multitude of scheduling and balancing parameters. When a single VM becomes overloaded while numerous empty VMs exist within the cloud network, redistributing workloads from overloaded to underutilized VMs is advantageous [12]. In cloud environments, it might be challenging to calculate all possible mappings of task resources, and even more to identify optimal mapping. Therefore, an effective task

distribution method is needed to schedule tasks in a way that prevents VMs from becoming excessively overloaded or underloaded [13]. The major contributions of the proposed DELICIOUS approach are as follows,

- This research proposes a novel, Dynamic Efficient Load Balancing in Cloud using kookaburra Infused pelican Optimization for virtUal Server (DELICIOUS) framework is to provide high quality services to customers in cloud computing applications.
- Initially, a task scheduling process is implemented to assign deadlines and execution times to tasks, while a load balancing process ensures workload migration in case of VM violations, due to that maintains load balance in the cloud environment.
- The performance evaluation of the DELICIOUS framework encompasses a comprehensive assessment, focusing on key metrics such as throughput, execution time, latency, waiting time, computational complexity, and computational cost.

The remainder of the study is organized as follows. Section II contains related papers with updates on recent research and descriptions of load balancing and task scheduling. Additional details about the proposed framework are provided in Section III. Section IV presents the experimental results of the proposed framework. The conclusions and further research are presented in Section V.

2. LITERATURE SURVEY

This section contains a summary of the literature has been discussed in this research. The concept of load balancing is initially discussed, along with its established model, metrics, and algorithms. Subsequently, it discusses recent literature on Load Balancing, presenting suggested algorithms by researchers and comparing their proposals with existing algorithms in the field.

In 2020, Devaraj, A.F.S., et al. [14] suggested a load balancing method with Firefly and Advanced Multi-Objective Particle Ensemble Optimization (FIMPSO). The simulation results revealed that the FIMPSO algorithm achieved the most efficient outcomes, with a shortest common response time of 13.58ms, surpassing all other comparable techniques. It provides the highest CPU utilization of 98%, memory utilization of 93%, dependability rating of 67%, throughput of 72%, and maximum make span of 148% respectively.

In 2021, Park, J., et al. [15] suggested GRAF, a proactive resource allocation technique utilizing graph neural networks to minimize overall CPU usage while assigning latency Service Level Objectives (SLOs). In comparison to autoscaling approaches, GRAF can achieve SLO latency with up to 19% of CPU consumption, according to experiments conducted using a number of publicly available benchmarks. Additionally, GRAF efficiently handles traffic spikes with 36% fewer resources than Kubernetes autoscaling and achieves faster latency convergence, up to 2.6 times respectively.

In 2022, Latchoumi, T.P., and Parthiban, L. [16] suggested to provide efficient resource scheduling in Cloud Computing (CC) scenarios with an essentially Quasi Opposite Dragonfly Algorithm technique for Load Balancing (QODA-LB). QODA-LB aimed to reduce task execution costs and times while ensuring an even distribution of workload across all VMs in the CC system. Simulation results demonstrated superior performance over leading methods, achieving optimal load balancing efficiency.

In 2022, Bal, P.K., et al. [17] suggested RATS-HM, a hybrid machine learning approach that blends safe resource allocation in a cloud computing context with effective task scheduling. Through simulations across different setups, RATS-HM shows the effectiveness which is compared to other state-of-the-art methods.

In 2023, Al Reshan, M.S., et al. [18] suggested an approach to load balancing in cloud computing which is fast and globally optimal. The suggested approach fused Gray Wolf Optimization with Particle Swarm Optimization (GWO-PSO) to achieve the advantages of both global optimization and quick convergence. By utilizing the GWO-PSO algorithm, this technique improves PSO convergence to 97.253% while reducing 12% on overall response times.

In 2023, Ramya, K., and Ayothi, S., [19] suggested Hybrid Whale and Dingo Optimization Algorithm (HDWOA-LBM) approach for cloud computing environments. This approach imitates a dingo's hunting behavior, optimizes the assignment of tasks to the appropriate virtual computer. The simulation experiments of HDWOA-LBM achieve a significant improvement in throughput of 21.28%, reliability of 25.42%, make span of 22.98%, and resource allocation of 20.86% for intelligent load balancing.

In 2024, Khaleel, M.I., [20] suggested RASA and dynamic task scheduling approach to balance the load in a cloud computing system. The RASA approach categorizes the tasks based on critical parameters using a task classification model. This approach resulted in reductions in latency overhead of 9%, processing time of 14%, workload imbalance of 15%, energy consumption of 19%, and idle time of 26%, along with improvements in resource availability of 22%, resource efficiency of 27%, and throughput of 32% respectively.

In order to address the above issues with cloud platforms, this research offers an agile task scheduling strategy that provides priority to crucial task characteristics including deadlines and durations which significantly affects the Quality of Service (QoS). Through meticulous scheduling and adherence to VM constraints, the algorithm ensures a well-balanced workload distribution across the cloud infrastructure.

3. DYNAMIC EFFICIENT LOAD BALANCING IN CLOUD USING KOOKABURRA INFUSED PELICAN OPTIMIZATION FOR VIRTUAL SERVER

In this section, a novel Dynamic Efficient Load Balancing in Cloud using Kookaburra-Infused Pelican

Optimization for Virtual Server (DELICIOUS) framework is proposed to deliver high-quality services to clients in Cloud Computing applications. The proposed DELICIOUS approach performs two main processes such as the task scheduling process, which is responsible for assigning deadlines and completion times to tasks, and the load balancing process, which manages the migration of workloads within Virtual Machine (VM) breach case to maintain load balance in cloud environment. Initially, the tasks requested by the user will be fed into the data center controller. These tasks are then passed to the load balancer, which uses a learning agent based on the Kookaburra-Pelican Hybrid Optimization Algorithm (HK-POA) to determine appropriate values and allocate tasks for machines in virtual environments suitable for cloud computing environments. The VM Manager then schedules tasks on the VM based on workload conditions such as underload and overload identified in the environment, using the migration process. The general block diagram of the proposed DELICIOUS framework is illustrated in Figure 1.

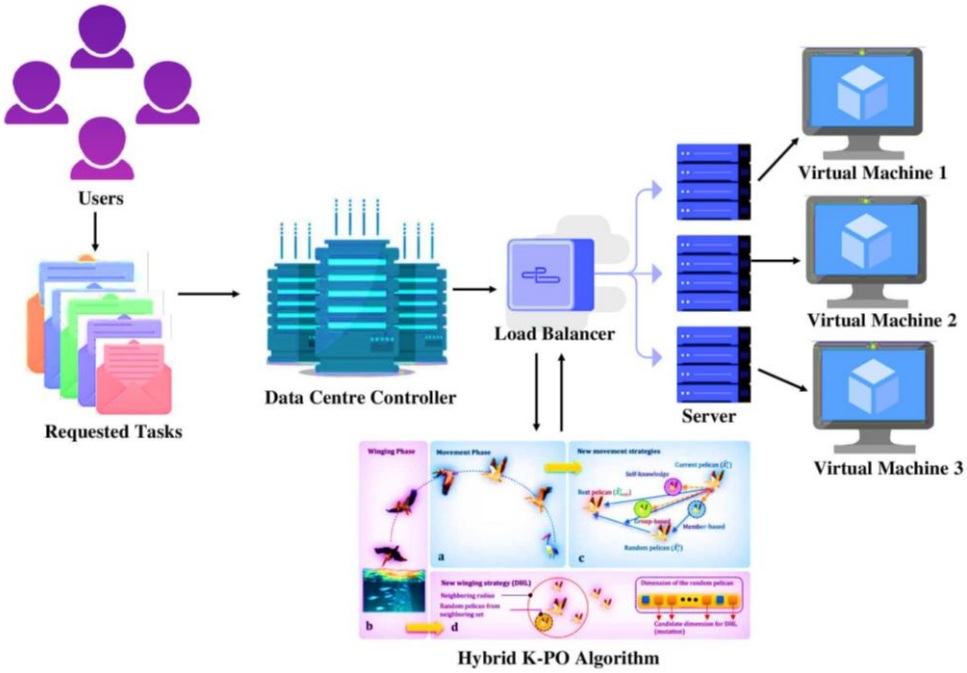


Figure 1. Proposed DELICIOUS Framework

3.1. Task classification

In task classification, algorithm 1 describes the process of adding tasks to the queue, with the primary determining factor being the task deadline. The underlying cloud hardware can accommodate a predetermined number (v) of virtual machines (VMs) to host tasks that users submit to the cloud scheduler. Dispatching these tasks presents challenges due to parameters such as τ_{cpu}, τ_{mem}, τ_{eed}, τ_{ded}, and τ_{ic} characterizing each task, alongside each VM being associated with ν_{cpu} and ν_{mem}, denoting CPU and memory resources, respectively. Each task demands a portion of the VMs' finite resources, contingent upon its size and duration. In response to this requirement, incoming workloads are classified into three distinct queues which is ϱ_{cpu}, ϱ_{mem}, and ϱ_{io}. For example, ϱ_{cpu} encompasses tasks necessitating intensive CPU utilization, ϱ_{mem} comprises those requiring substantial memory utilization, and ϱ_{io} contains tasks with prolonged durations.

Based on its parameters, each tasks speed is determined using R=(τ/S). The task category with the greatest rate is chosen as τC=max [R_{cpu}, R_{mem}, R_{io}]. After being accepted, tasks are initiated and categorized into three groups which is τC_{mem}, τC_{cpu}, and τC_{io}.. Then, the outer loop makes sure that each authorized task is finished on schedule. The execution time of each task is calculated, ensuring adherence to its Service Level Agreement (SLA), including its deadline. A queue, ϱ_{queue}, is employed to organize all accepted tasks meeting their deadlines. The loop verifies that tasks are arranged suitably according to their requirements by continuously assessing the memory, CPU, and I/O speeds of each task. Tasks demanding significant CPU utilization are classified first, followed by those necessitating higher memory utilization, and finally, those requiring longer durations are separated.

3.2. Load Balancing Via Hybrid K-POA

This section commences by elucidating the inspiration and theoretical foundation behind the proposed Hybrid Kookaburra-Pelican Optimization Algorithm (HK-POA), followed by a mathematical modeling of its implementation steps for solving optimization problems. Kookaburra optimization is an iterative method for solving problems in the realm of problem solving that uses stochastic search to produce effective solutions to optimization problems. Based on the collective population matrix of Kookaburras, the Kookaburra Optimization Algorithm (KOA) is represented mathematically by equation (1). According to equation (2), kookaburras' starting locations during KOA deployment are chosen at random.

$$X = \begin{bmatrix} X_1 \\ . \\ . \\ . \\ X_i \\ . \\ . \\ . \\ X_N \end{bmatrix}_{N\times m} = \begin{bmatrix} x_{1,1} & \cdots & x_{1,d} & \cdots & x_{1,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{i,1} & \cdots & x_{i,d} & \cdots & x_{i,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{N,1} & \cdots & x_{N,d} & \cdots & x_{N,m} \end{bmatrix}_{N\times m} \quad (1)$$

$$x_{i,d} = lb_d + r(ub_d - lb_d) \quad (2)$$

The general KOA matrix, represented by X includes the search space. N is the total number of kookaburras, with m optimal variables. A random number inside the interval [0, 1] is shown, and the variables ub_d and lb_d, respectively, indicate the lower and upper bounds of the ith decision variable through the variable r. Equation (3) demonstrates that a vector can be used to describe the collection of objective function values that were assessed for the task.

$$F = \begin{bmatrix} F_1 \\ . \\ . \\ . \\ F_i \\ . \\ . \\ . \\ F_N \end{bmatrix}_{N\times 1} = \begin{bmatrix} F(X_1) \\ . \\ . \\ . \\ F(X_i) \\ . \\ . \\ . \\ F(X_N) \end{bmatrix}_{N\times 1} \quad (3)$$

The vector F, where F_i identifies the objective function evaluated corresponding to the ith kookaburra, specifies the collection of objective functions assessed in this scenario. The prey set that each kookaburra has access to is then ascertained by comparing the values of the objective functions, as indicated by equation (4).

$$CP_i = \{X_k : F_k < F_i and\ k \neq i\}, where\ i = 1, 2, \ldots\ldots, N\ and\ k \in \{1, 2, \ldots\ldots, N\} \quad (4)$$

In this case, the set of possible prey that the ith kookaburra can approach is shown by CP_i. F_k represents the objective function value in this instance, and X_k represents the kookaburra with an objective function value greater than the ith kookaburra. Using equation (5), which mimics the kookaburra's path toward the prey while hunting, the bird's new location is ascertained. The kookaburra in concern will relocate to this new site if the objective function's value there rises, as demonstrated by equation (6).

$$x_{i,d}^{p1} = x_{i,d} + r\left(SCP_{i,d} - I.x_{i,d}\right), i = 1, 2, \ldots\ldots, N, \quad and\ d = 1, 2, \ldots\ldots, m \quad (5)$$

$$X_i = \begin{cases} x_i^{p1}, & F_i^{p1} < F_i \\ X_i, else \end{cases} \quad (6)$$

F_i^{p1} represents the functional objective value, $x_{i,d}^{p1}$ represents the new recommended position of the ith kookaburra based on the first step of the KOA and $x_{i,d}^{p1}$ represents the kookaburra's dimension d. Furthermore, r is a random number selected from 0 to 1 based on a normal distribution. The $SCP_{i,d}$ represents the d-th dimension of the prey chosen for the ith kookaburra, where i is a randomly chosen number from the set {1, 2}. N stands for the total number of kookaburras, and m for the number of decision variables.

The KOA technique imitates the behaviour of kookaburras around hunting sites by utilizing equation (7) to determine an arbitrary location. Equation (8) indicates that the previous position will be replaced if the new location for each kookaburra raises the value of the objective function.

$$x_{i,d}^{p2} = x_{i,d} + (1 - 2r).\frac{(ub_d - lb_d)}{t}, i = 1, 2, \ldots\ldots, N, d = 1, 2, \ldots\ldots, m, and\ t = 1, 2, \ldots\ldots, T \quad (7)$$

$$X_i = \begin{cases} x_i^{p2}, & F_i^{p2} < F_i \\ X_i, else \end{cases} \quad (8)$$

The function's target value in this instance is F_i^{p2}, the dth dimension is represented by x_i^{p2}, and the ith kookaburra's proposed new location is x_i^{p2}, which is based on the second phase of KOA. Furthermore, T denotes the algorithm's maximum iteration count, and t denotes the algorithm's iteration counter.

The proposed PO algorithm is a population-based approach where pelicans constitute the members of this population. Initially, the extracted features are randomly initialized within the lower and upper bounds of the problem using Equation (9).

$$M_{p,q} = h_q + rand \cdot \left(k_q - h_q\right), p = 1, 2, \ldots, X, q = 1, 2, \ldots y \quad (9)$$

This equation gives the value of the variable qth, which is decided by the candidate solution pth, represented by $M_{p,q}$. The lower and upper bounds, represented by h_q and k_q, respectively, express the total population. The mathematical expression for the pelican's strategy as it moves towards its prey is provided in equation (10).

$$M_{p,q}^{A_1} = \begin{cases} M_{p,q} + rand \cdot \left(A_q - G \cdot M_{p,q}\right), & E_a < E_p \\ M_{p,q} + rand \cdot \left(M_{p,q} - A_q\right), & else \end{cases} \quad (10)$$

This equation defines $M_{p,q}^{A_1}$, indicating the updated status of the pth pelican in the qth dimension after phase 1. The variable G takes on a random value of either 1 or 2. A_q represents the prey's position in the qth dimension, while E_a denotes its objective function value. Hence, parameter G significantly influences PO's exploration by accurately selecting features. This behavior, reminiscent of pelicans during hunting, is mathematically simulated in the equation (11).

$$M_{p,q}^{A_2} = M_{p,q} + D \cdot \left(1 - \frac{t}{T}\right) \cdot (2 \cdot rand - 1) \cdot M_{p,q} \quad (11)$$

In this equation, $M_{p,q}^{A_2}$ represents the updated state of the pth pelican in the qth dimension after the time frame. D(1-t/T) gives the neighbourhood radius of $M_{p,q}$ with D fixed at 0.2. The coefficient D(1-t/T) is pivotal for PO's exploitability, as it brings the algorithm closer to the global optimal solution. Initially, during the early iterations, this coefficient has a larger value, resulting in a broader

exploration around each member. However, as the algorithm progresses through subsequent phases, the pelican positions are adjusted, and the most promising features are identified. Eventually, the algorithm stops and produces the best features when the maximum number of iterations is reached or the convergence condition is satisfied. The algorithm 1 shows the steps for HK-POA which is given below.

Initialize Kookburra population X_k and pelican population X_p.

Initialize iteration count $t = 0$.

Exploration Phase of Kookaburra

For each Kookaburra individual x_{ki} in X_k:

Update the position using:

$$x_{ki} = x_{ki} + \alpha.rand(0,1).StepSize$$

Exploitation phase of Pelican

For each Pelican individual x_{pi} in X_p:

Update the position using:

$$x_{pi} = x_{pi} + \beta.rand(0,1).\nabla f(x_{pi})$$

Combine Kookaburra and Pelican populations: $X = X_k \cup X_p$

Select individuals for the next generation based on fitness.

Increment t.

If termination conditions are not met stop;

Otherwise, go to step 3.

---

4. RESULTS AND DISCUSSION

In this section, the simulation setup and outcomes of the proposed DELICIOUS method employing various validation and assessment studies are discussed. In the experimental evaluation of the proposed DELICIOUS method, classes from the extended CloudSim toolkit are utilized for modelling and simulation of cloud systems. CloudSim Simulator allows to set up a virtualized environment with on-demand resource provisioning. Furthermore, cloud services and related applications may be more easily simulated, modelled, and tested. A few key factors taken into account are throughput, execution time, latency, waiting time, computational complexity, and computational cost which is then compared to the number of available tasks.

4.1. Performance Evaluation

The efficiency of the proposed DELICIOUS approach was assessed using critical performance indicators for the first time, including throughput, latency, execution time, delay, computational complexity, and computation time and fundamental methods of implementation, such as RATS-HM, GRAF, and QoDA-LB. Figure 2 displays the throughput attained for multiple tasks using the proposed DELICIOUS approach along with comparison methodologies. The proposed DELICIOUS design, performs better than the other three systems when processing workloads out of 500 tasks, with throughputs of 865 Kbps, 943.4 Kbps, and 984.6 Kbps respectively.

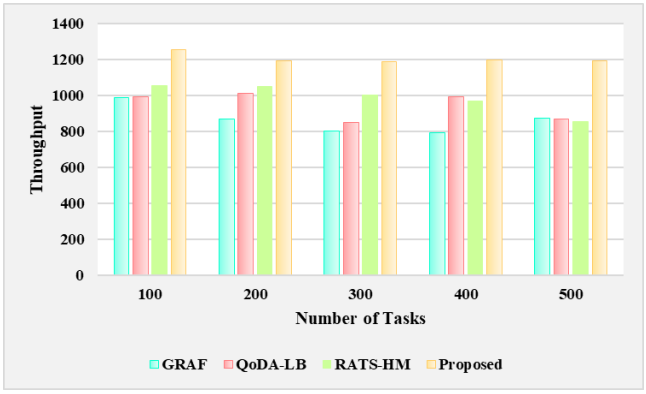


Figure 2. Throughput Vs Number of Tasks

These outcomes shows that the proposed DELICIOUS approach is effective and that it is feasible to take advantage of integrated HK-POA benefits. Through a comprehensive examination of numerous factors, this method delivers optimal throughput regardless of the workload entering the cloud environment by allocating incoming tasks to the right VM's.

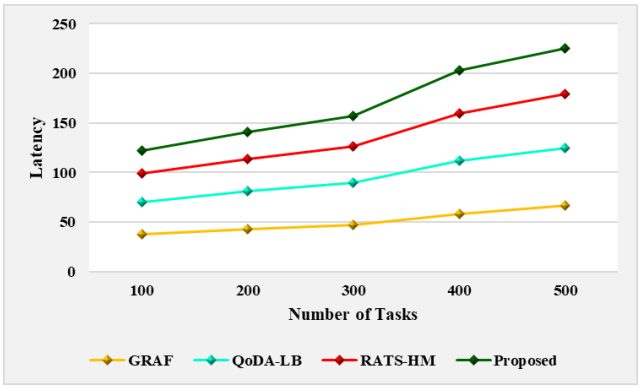


Figure 3. Latency Vs Number of Tasks

Figure 3 illustrates the delay that the proposed DELICIOUS approach, as well as the previously stated GRAF, QoDA-LB, and RATS-HM techniques, encountered for varying numbers of tasks. The performance of the proposed DELICIOUS method is superior to the methods discussed, as shown by the latency trends for varying task counts.

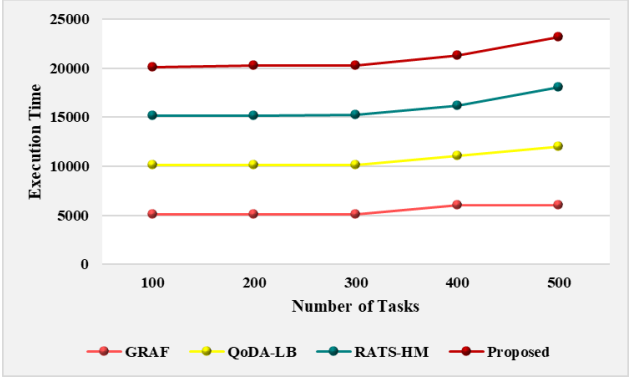


Figure 4. Execution Time Vs Number of Tasks

This decrease in latency is credited to the scheme's adeptness and consistency in dynamically applying diverse load balancing constraints during task scheduling. Moreover, it excels in effectively allocating tasks to suitable VMs, thus averting situations of under-utilization or over-utilization in the cloud environment.

Furthermore, Figure 4 and Figure 5 displays the proposed DELICIOUS method for various workloads together with the latency and execution durations for the evaluated GRAF, QoDA-LB, and RATS-HM approaches. Particularly, the proposed DELICIOUS scheme exhibits reduced execution time and waiting time relative to the baseline approaches under varying task.

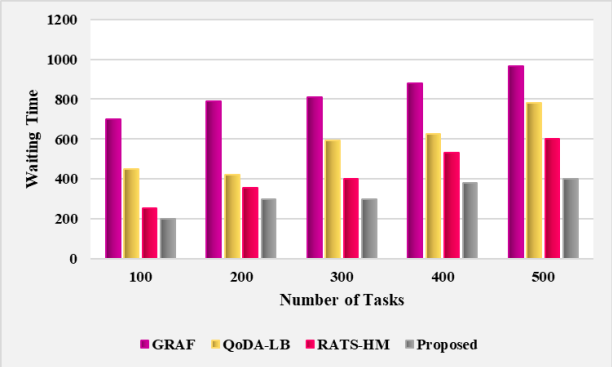


Figure 5. Waiting Time Vs Number of Tasks

By integrating the multi-dimensional HK-POA, the scheme explores essential factors for assigning incoming tasks to suitable VMs within the cloud environment. This dynamic exploration facilitated by the proposed DELICIOUS scheme effectively decreases task waiting times by dynamically adjusting the load balancing rate to meet the required criteria.

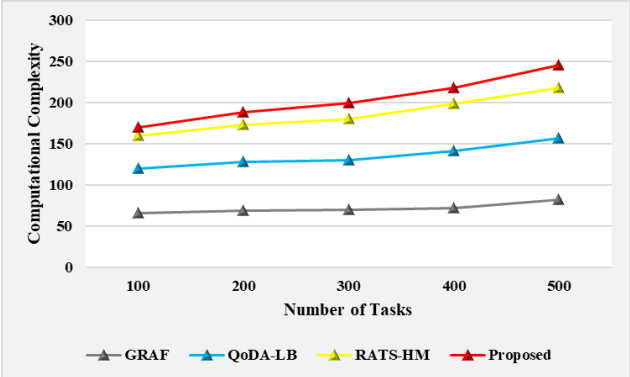


Figure 6. Computational Complexity Vs Number of Tasks

Figures 6 and 7 display the computational complexity, cost, and techniques of comparison of the suggested system for various activities. The efficacy of the proposed DELICIOUS system is assessed in this part. The basic load balancing techniques, such as the GRAF, QoDA-LB, and RATS-HM schemes, concentrate on task efficiency, computing complexity, and cost.

By including constraints and multi-objective optimization elements into the load balancing process, the proposed DELICIOUS strategy combines processing complexity and cost, independent of the number of operations. Moreover, it includes diverse load balancing parameters to dynamically assign tasks to different VMs throughout the allocation process.

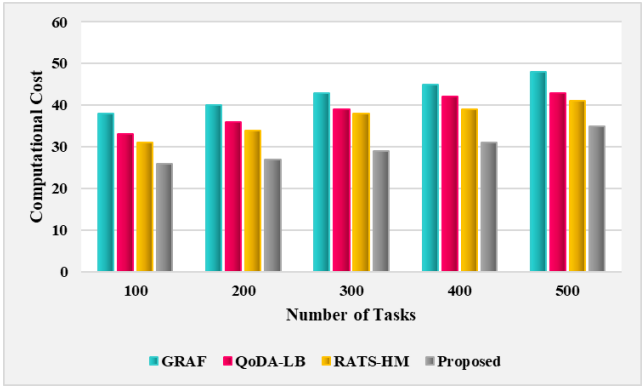


Figure 7. Computational Cost Vs Number of Tasks

5. CONCLUSION

This paper proposed a novel Dynamic Efficient Load Balancing in Cloud using kookaburra Infused pelican Optimization for virtUal Server (DELICIOUS) is developed for Effective load balancing system in cloud computing. The proposed DELICIOUS technique is validated by using the Cloud Simulator (CloudSim) to providing consumers with the best services for applications using cloud computing. Furthermore, the performance of the proposed DELICIOUS technique is evaluated in terms of the parameters such as throughput, execution time, latency, waiting time, computational complexity, and computational cost. The DELICIOUS technique achieves a better throughput of 1206.6 Kbps whereas, the GRAF, QoDA-LB, and RATS-HM technique attains 865 Kbps, 943.4 Kbps, and 984.6 Kbps respectively.

CONFLICTS OF INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

FUNDING STATEMENT

Not applicable.

ACKNOWLEDGEMENTS

The author would like to express his heartfelt gratitude to the supervisor for his guidance and unwavering support during this research for his guidance and support.

REFERENCES

[1] F. Alqahtani, M. Amoon, and A.A. Nasr, “Reliable scheduling and load balancing for requests in cloud-fog computing”, *Peer-to-Peer Networking and Applications*, vol. 14, no. 4, pp.1905-1916, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[2] K. Balaji, P.S. Kiran, and M.S., Kumar, “An energy efficient load balancing on cloud computing using adaptive cat swarm optimization”, *Materials Today: Proceedings*, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[3] F.M. Talaat, M.S. Saraya, A.I. Saleh, H.A. Ali, and S.H. Ali, “A load balancing and optimization strategy (LBOS) using reinforcement learning in fog computing environment”, *Journal of Ambient Intelligence and Humanized Computing*,

vol. 11, no. 11, pp.4951-4966, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[4] S. Sagar, M. Ahmed, and M.Y. Husain, "Fuzzy Randomized Load Balancing for Cloud Computing", *In International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*, pp. 18-29, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[5] N. Arivazhagan, K. Somasundaram, D. Vijendra Babu, M. Gomathy Nayagam, R.M. Bommi, G.B. Mohammad, P.R. Kumar, Y. Natarajan, V.J. Arulkarthick, V.K. Shanmuganathan, and K. Srihari, "Cloud-internet of health things (IOHT) task scheduling using hybrid moth flame optimization with deep neural network algorithm for E healthcare systems", *Scientific Programming*, 2022, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[6] A. Asghari, and M.K. Sohrabi, "Combined use of coral reefs optimization and reinforcement learning for improving resource utilization and load balancing in cloud environments", *Computing*, vol. 103, no. 7, pp.1545-1567, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[7] H. Mahmoud, M. Thabet, M.H. Khafagy, and F.A. Omara, "An efficient load balancing technique for task scheduling in heterogeneous cloud environment", *Cluster Computing*, vol. 24, no. 4, pp. 3405-3419, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[8] Z. Miao, P. Yong, Y. Mei, Y. Quanjun, and X. Xu, "A discrete PSO-based static load balancing algorithm for distributed simulations in a cloud environment", *Future Generation Computer Systems*, vol. 115, pp.497-516, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[9] J.R. Adaikalaraj, "Load Balancing In Cloud Computing Environment Using Quasi Oppositional Dragonfly Algorithm", *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, no. 10, pp. 3256-3273, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[10] S. Dhahbi, M. Berrima, and F.A. Al-Yarimi, "Load balancing in cloud computing using worst-fit bin-stretching", *Cluster Computing*, vol. 24, no. 4, pp. 2867-2881, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[11] S. Afzal, and G. Kavitha, "Load balancing in cloud computing–A hierarchical taxonomical classification", *Journal of Cloud Computing*, vol. 8, no. 1, pp. 22, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[12] U. K. Jena, P. K. Das, and M. R. Kabat, "Hybridization of meta-heuristic algorithm for load balancing in cloud computing environment", *Journal of King Saud UniversityComputer and Information Sciences*, vol. 34, no. 6, pp.2332- 2342, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[13] E. H. Houssein, A. G. Gad, Y. M. Wazery and P. N. Suganthan, "Task scheduling in cloud computing based on meta-heuristics: review, taxonomy, open challenges, and future trends", *Swarm and Evolutionary Computation*, vol. 62, pp.100841, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[14] A.F.S. Devaraj, M. Elhoseny, S. Dhanasekaran, E.L. Lydia, and K. Shankar, "Hybridization of firefly and improved multi-objective particle swarm optimization algorithm for energy efficient load balancing in cloud computing environments", *Journal of Parallel and Distributed Computing*, vol. 142, pp.36-45, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[15] J. Park, B. Choi, C. Lee, and D. Han, GRAF: A graph neural network based proactive resource allocation framework for SLO-oriented microservices", *In Proceedings of the 17th International Conference on emerging Networking Experiments and Technologies*, pp. 154-167, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[16] T.P. Latchoumi, and L. Parthiban, "Quasi oppositional dragonfly algorithm for load balancing in cloud computing environment", *Wireless Personal Communications*, vol. 122, no. 3, pp.2639-2656, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[17] P.K. Bal, S.K. Mohapatra, T.K. Das, K. Srinivasan, and Y.C. Hu, "A joint resource allocation, security with efficient task scheduling in cloud computing using hybrid machine learning techniques", *Sensors*, vol. 22, no. 3, pp.1242, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[18] M.S. Al Reshan, D. Syed, N. Islam, A. Shaikh, M. Hamdi, M.A. Elmagzoub, G. Muhammad, and K.H. Talpur, "A fast converging and globally optimized approach for load balancing in cloud computing", *IEEE Access*, vol. 11, pp.11390-11404, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[19] K. Ramya, and S. Ayothi, "Hybrid dingo and whale optimization algorithm-based optimal load balancing for cloud computing environment", *Transactions on Emerging Telecommunications Technologies*, vol. 34, no. 5, p.e4760, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[20] M.I. Khaleel, "Region-aware dynamic job scheduling and resource efficiency for load balancing based on adaptive chaotic sparrow search optimization and coalitional game in cloud computing environments", *Journal of Network and Computer Applications*, vol. 221, pp.103788, 2024. [CrossRef] [Google Scholar] [Publisher Link]

AUTHORS

G. Saranya received her B.E degree in Computer Science and Engineering from Anna University, Chennai and M.E degree in Computer Science and Engineering from Hindustan University, Chennai. She started her career as an Assistant Professor and has 9 years and 6 months of experience. Currently she is working as an Assistant Professor in S.A. Engineering College, Chennai. Her research interests include Deep Learning and Cloud Computing. She is a lifetime member of ISTE.

G Belshia Jebamalar currently working as Assistant professor in Department of computer science in SA Engineering college, Poonamallee, Thiruverkadu, Tamil Nadu 600077 India

Chukka Santhaiah I am currently associated with Sri Venkateswara College of Engineering, Tirupati. My current responsibilities include coordinating all the departmental operations, delivering the lectures on time, conducting the class assignments and carrying out necessary evaluation and organizing mid-sem exams, class assignments along with external and internal lab exams & evaluating the same. I am also responsible for conducting seminars, conferences, workshops. My academic credentials include a Doctor of Philosophy (Ph.D.) (CSE) from Sri Venkateswara University College of Engineering, India, Master of Technology (M.Tech) (CS) from Rajiv Gandhi College of Engineering, India, Bachelor of Technology (B. Tech) in CSE from Rajiv Gandhi College of Engineering, India. Proven record of13 +patents 3granted, and one text book published lambert publications Submitted research proposals for DST and SERB. Applied for IP award and DST Inspire award. Research Areas: Image Processing, Bioinformatics, Computer Networks, Machine Learning, Artificial Intelligence, and Data Science.

Arrived: 05.07.2024
Accepted: 10.08.2024

International Journal of Data Science and Artificial Intelligence (IJDSAI)
Volume 02, Issue 04, July - August (2024)

RESEARCH ARTICLE

IN-DEPTH EXPLORATION AND COMPARATIVE ASSESSMENT OF CUTTING-EDGE ALGORITHMS FOR IMPULSE NOISE ATTENUATION IN CORRUPTED VISUAL DATA

S. Prathiba [1, *] and B. Sivagami [2]

[1] Department of Computer Science & Research Centre, S.T. Hindu College, Nagercoil 629002, Affiliated to Manonmaniam Sundaranar University, Abishekapatti, Tirunelveli 627012, Tamil Nadu, India
[2] Department of Computer Science & Application, S.T. Hindu College, Nagercoil 629002, Affiliated to Manonmaniam Sundaranar University, Abishekapatti, Tirunelveli 627012, Tamil Nadu, India.

[*]*Corresponding e-mail: prathibasuyambu26@gmail.com*

Abstract – Image denoising is a vital process in image pre-processing, particularly for applications focused on image-based objectives. This process, which occurs during image acquisition and transmission, is crucial for enhancing image quality to facilitate subsequent analysis by medical image processing algorithms. Given its importance in improving medical images, image denoising has become a prominent research focus. This paper explores the latest advancements in denoising techniques specifically tailored for magnetic resonance imaging (MRI), offering a detailed examination of their publication details, underlying methodologies, strengths, limitations, and accuracy metrics, including peak signal-to-noise ratio (PSNR). The study provides a thorough review of contemporary denoising strategies proposed by researchers such as Taherkhani et al., Zhang et al., Yuan et al., and Chen et al., among others, presenting an in-depth survey of current denoising algorithms. Understanding these methods is critical for selecting robust denoising techniques capable of mitigating artifacts like salt-and-pepper noise, which is essential for effective medical image segmentation. This paper aims to provide valuable insights into denoising methodologies, thereby advancing MRI image processing in the medical domain.

Keywords *–Image Denoising, Magnetic Resonance Imaging (MRI), Medical Image Processing, Peak Signal-to-Noise Ratio (PSNR), Noise Reduction Algorithms.*

1. INTRODUCTION

Image denoising entails the reduction of noise in digital images, which can originate from various sources such as sensor limitations, transmission errors, or environmental factors [1]. A common type of noise is 'Salt and Pepper noise,' characterized by the presence of randomly distributed bright and dark pixels, resembling grains of salt and pepper, hence its name. Also known as impulse noise, Salt and Pepper noise can be introduced through multiple mechanisms, including transmission errors in digital images, electrical interference during image acquisition, or defects in image sensors [2]. This type of noise can significantly degrade image quality and impair its visual integrity. The noisy pixels appear as outliers in comparison to the surrounding pixels, resulting in a disruptive effect on the overall image [3]. Therefore, it is essential to remove or mitigate Salt and Pepper noise to restore the image to its original quality.

Various techniques are available for mitigating salt and pepper noise in images. A widely used method is the median filter, which substitutes each noisy pixel with the median value of its neighboring pixels. This technique is effective in reducing salt and pepper noise while preserving the image's edges and details. Another advanced method is the adaptive median filter, which dynamically adjusts the window size of the median filter according to the noise characteristics. This adaptability enhances the filter's performance in handling images with varying levels of noise [4].

Other noise reduction techniques include mean filtering, Gaussian filtering, and bilateral filtering. These approaches apply spatial filters to smooth out noise while striving to maintain essential image features. In addition to these conventional methods, more sophisticated techniques leveraging machine learning and deep learning—such as convolutional neural networks (CNNs)—have emerged for addressing salt and pepper noise [5]. These advanced methods utilize artificial intelligence to improve the image restoration process.

The choice of an effective noise removal technique is influenced by the specific characteristics of the noise, the desired degree of noise reduction, and the need to balance noise suppression with the preservation of image details.

Frequently, multiple methods must be tested to identify the optimal solution for a given image. This paper offers a comprehensive survey of image denoising methodologies, aiming to explore and assess these various approaches

2. LITERATURE REVIEW ON DENOISING TECHNIQUES

This survey paper examines eleven research studies focused on image denoising with an emphasis on salt and pepper noise. It includes detailed information such as author details, objectives, the principal algorithm used, descriptions of supporting algorithms, database names, and a comprehensive analysis of the merits and limitations of each study.

Taherkhani et al. [6] developed an algorithm utilizing Radial Basis Functions (RBFs) interpolation to estimate the intensities of corrupted pixels based on their neighboring pixels. This approach begins by estimating the intensity values of noisy pixels in the corrupted image using RBFs, followed by a smoothing process. The algorithm was tested on four standard 8-bit grayscale images: 'Boat,' 'Peppers,' 'Gold Hill,' and 'Barbara.' A key advantage of this method is its ability to restore images with enhanced visual quality, better edge definition, and preserved texture details. Additionally, the algorithm does not require parameter tuning through trial and error to achieve optimal results. However, a notable drawback is its ineffectiveness in addressing Gaussian and Speckle noise. Furthermore, excessive smoothing may lead to blurring, which can adversely affect texture areas.

Zhang et al. [7] introduced a data-driven algorithm for impulse noise removal using an Iterative Scheme-Inspired Network (ISIN). This network shifts the primary effort from the online optimization phase to a preliminary offline training stage, allowing it to be applied to new data using the learned parameters. The Berkeley Segmentation Dataset was utilized for testing. The advantage of this network lies in its efficient noise reduction capability through a straightforward iterative scheme. However, a drawback is that excessive smoothing may cause blurring in texture areas. Additionally, the network does not address the handling of multiple datasets.

Yuan et al. [8] proposed a model in the domain of regularization-based image processing featuring a novel sparse optimization technique known as l0TV-PADMM. This approach addresses the Total Variation (TV)-based restoration problem with `l0-norm data fidelity. Test images of size 512 × 512 were utilized in this model, and the entire implementation is conducted in MATLAB. The model's advantage lies in its improved handling of image denoising and deblurring in the presence of impulse noise. However, a limitation of this technique is its inefficiency with high-resolution images. Additionally, since the model is not implemented in C++, it exhibits slower performance.

Chen et al. [9] putforth an adaptive sequentially weighted median filter (ASWMF) designed for images affected by impulse noise. This method incorporates a noise detector that utilizes the 3σ principle of normal distribution alongside local intensity statistics. The ASWMF applies a sequentially weighted median filter with an adaptive neighborhood size, where the weights are determined based on the spatial distances from the central noisy pixel. The input datasets for testing include SET12, BSD68, and medical images. The advantage of the ASWMF lies in its superior performance compared to state-of-the-art filters in managing impulse noise, as well as its significantly reduced computational time. However, the filter faces challenges in real-time denoising applications and, being implemented in MATLAB rather than C++, exhibits slower processing speed.

Sonali et al. [10] presented a noise removal and contrast enhancement algorithm specifically for fundus images. This technique integrates various filters with the Contrast Limited Adaptive Histogram Equalization (CLAHE) method to address both denoising and enhancement issues in color fundus images. Initially, the fundus RGB image is decomposed into its red, green, and blue channels. Different filtering techniques, along with CLAHE, are applied to each channel to reduce noise and enhance contrast. The processed channels are then combined to produce an improved RGB fundus image. The technique uses RGB fundus images of size 605 x 700 from the STARE database for simulation. The advantages of this method include effective noise removal and contrast enhancement in fundus images, with performance metrics surpassing those of state-of-the-art methods. However, the technique is limited to the medical domain and specifically designed for fundus images. Additionally, it does not address the needs of color medical images outside this scope.

Jin et.al. [11] developed an Image Recovery Method (IRM) leveraging deep convolutional neural networks for the removal of impulse noise. This denoising framework employs two deep CNNs: a classifier network and a regression network. The classifier network is trained to identify noisy pixels within an image, while the regression network is designed to reconstruct the denoised image. The input images used for testing include Lena, Cameraman, Barbara, and Boat. The primary advantage of this method is its superior denoising performance. However, it is hindered by its high running time and increased computational complexity.

Zhang et al. [12] put forth an Exemplar-Based Image De-Noising Algorithm (EIDA) that demonstrates significant potential for image restoration. The method incorporates a parameterized surrogate of the l0 norm to address both low-rank and sparse constraints, with analytic solutions provided for the associated optimization problems. The algorithm is evaluated using Lansel's benchmark dataset, focusing on grayscale images. The key advantage of EIDA is its superior performance in image restoration. However, the algorithm has limitations, including its applicability to only a single dataset and reduced efficiency with other types of images.

Sheela et al. [13] presented an Adaptive Switching Modified Decision-Based Unsymmetric Trimmed Median Filter (ASMDBUTMF) designed for noise reduction in grayscale MRI images affected by salt and pepper noise. This method adaptively selects overlapping windows for the noise reduction process, specifically targeting medical MRI

images to minimize noise. The advantage of this technique is its potential use as a preprocessing step for scanning machines, enhancing robustness in noisy environments. However, the method is limited to MRI images and does not perform effectively with other types of images. Additionally, it does not address the removal of random noise types.

Li et al. [14] presented a method called the Densely Connected Network for Impulse Noise Removal (DNINR), which employs convolutional neural networks (CNNs) to learn pixel-distribution features from noisy images. This method is based on advanced non-linear learning and residual learning principles. The test images used include Barbara, Baboon, Boat, C-man, and Foreman. The primary advantage of DNINR is its superior performance in preserving edges and suppressing noise. However, the method encounters limitations when dealing with non-Gaussian noises, such as Poisson and Rician noise, and it suffers from high computational complexity.

Wang et al. [15] presented a method for detecting noise points in images using Fractional Differential Gradient (FDG), with an enhanced image denoising algorithm based on fractional integration. The method is evaluated using grayscale images, such as Lena. The advantage of this model is its effective noise removal while preserving image edge details. However, the model's performance is limited when applied to other types of images and is assessed with a relatively small number of test images.

Liu et al. [16] put forth a nonlinear Spline Adaptive Filter based on the Robust Geman-McClure Estimator (SAF-RGM). This algorithm employs a cost function derived from the Geman-McClure estimator and processes signals generated by a Gaussian process. The advantage of this filter is its superior stability in the presence of impulsive noise. However, the technique is characterized by high time consumption and significant computational complexity.

Golam Muktadir Mukti et al. [17] present a Mat Lab-based Noise Removal Technique (MNRT) for removing salt and pepper noise from brain MR image. The goodness of this technique is that this weighted median filter provides high quality images by removing salt and pepper noises. The drawback of this technique is that it loses its efficiency while working with the kernel size above three.

XuYan et al. [18] developed Unsupervised Image De-noising algorithm based on Generative Adversarial Networks (UIDGAN). The model employs perceptual loss and cycle-consistency loss to ensure consistency of content information which is considered it to be its shinning side. The drawback of this method is that it considers many parameters which in turn increases its complexity and processing time.

3. DISCUSSION AND ANNALYSIS

Table 1 reports the considered research papers about their author's name, publication, published year, and the core method that is implemented. The core method column indicates that the Densely Connected Network, Fractional Differential Gradient and Geman-McClure technologies dominated in the recent noise removal algorithm.

Table 1. Representation of publication information and core technology used

Author name	Publication	Year	Methodology Used
Taherkhani et al.	IET	2017	RBF
Zhang et al.	SPRINGER	2018	ISIN
Yuan et al.	IEEE	2018	*l0TV*-PADMM
Chen et al.	IEEE	2019	ASWMF
Sonali et al.	ELSEVIER	2019	CLAHE
Jin et.al.	ELSEVIER	2019	IRM
Zhang et al.	IEEE	2020	EIDA
Sheela et al.	ELSEVIER	2020	ASMDBUTMF
Li et al.	SPRINGER	2020	DNINR
Wang et al.	ELSEVIER	2020	FDG
Liu et al.	IEEE	2020	SAF-RGM
Mukti et al	IJRES	2022	MNRT
XuYan et al	JTPES	2024	UIDGAN

Table 2. Description of the merits and demerits of the reviewed methods

Method	Advantage	Disadvantage
Taherkhani et al. [6]	Higher visual quality	Gaussian and Speckle noises
Zhang et al. [7]	Noise reduction with the simple iterative scheme	Not tested for multiple datasets
Yuan et al. [8]	Problem of de-blurring is addressed in a better manner	Not efficient for high resolution images
Chen et al. [9]	Low computational time	Not supportive for applied for real time de-noising
Sonali et al. [10]	Remove noises and enhance contrast in fundus images	color medical images is not addressed
Jin et.al. [11]	Better denoising performance	Higher computational complexity.
Zhang et al. [12]	Image restoration is better	Not supportive for other type of images
Sheela et al. [13]	shows better Better robustness	Random noises cannot be removed
Li et al. [14]	Better performance on edge preservation and noise suppression	Non-Gaussian noises like Poisson noise and Rician noise cannot be supported
Wang et al. [15]	Preserves the details of image edges in a better manner	Evaluated only by the minimum quantity of test images.

Liu et al. [16]	Better performance against impulsive noise	High time consumption and high computational complexity.
Mukti et al. [17]	It provides high quality images by removing salt and pepper noises	It loses its efficiency while working with the kernel size above three
XuYan et al. [18]	The consistency of content information is maintained	It considers many parameters which in turn increases its complexity

Table 2 discusses the advantages and disadvantages of considered research papers. This analysis reveals that the recent advantages of the researches on denoising are detail preservation and edge structure preservation. Also, the recent papers in denoising field suffers based on the demerits like high computational complexity and incapable in high resolution images.

Table 3. PSNR analysis for 50% of noise pollution

Methods	PSNR (in db)
[6]	26.91
[7]	26.81
[8]	26.93
[9]	27.19
[10]	27.22
[11]	27.08
[12]	27.51
[13]	27.18
[14]	28.20
[15]	28.41
[16]	28.53
[17]	29.83
[18]	30.22

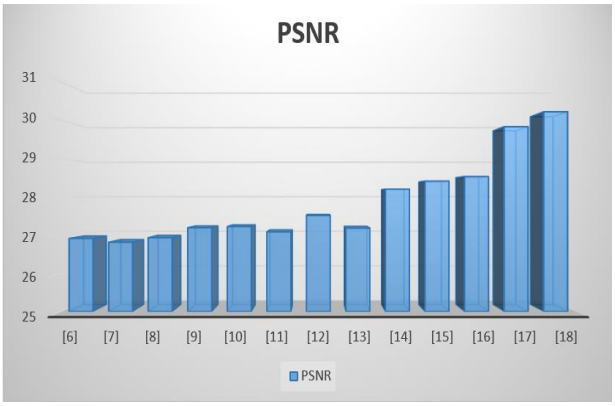


Figure 1. Representation of PSNR for 50% of noise corruption**.**

Table 3 and Figure 1 focus on the peak signal to noise ratio (PSNR) of the research papers on denoising. The authors Wang et al. and Liu et al. designed the two effective methods on requiring impulse noises which are proved by the PSNR results.

4. CONCLUSION

This survey paper examines recent advancements in noise reduction techniques, with a focus on various impulse noise suppression strategies. It provides an in-depth analysis of eleven denoising methods to elucidate the underlying principles of existing approaches. The research findings indicate that the methods developed by Wang et al. and Liu et al. demonstrate superior denoising performance, particularly in terms of PSNR. In contrast, the denoising technique proposed by Taherkhani et al. is found to be less effective compared to other methods. A thorough understanding of the noise characteristics and the degree of image degradation is crucial for selecting the most suitable filters and algorithms.

CONFLICTS OF INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

FUNDING STATEMENT

Not applicable.

ACKNOWLEDGEMENTS

The author would like to express his heartfelt gratitude to the supervisor for his guidance and unwavering support during this research for his guidance and support.

REFERENCES

[1] C. Cruz, A. Foi, V. Katkovnik and K. Egiazarian, “Nonlocality-Reinforced Convolutional Neural Networks for Image Denoising”, *IEEE*, vol. 25, no. 8, pp. 1216-1220, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[2] K.K.V. Toh, H. Ibrahim and M.N. Mahyuddin, “Salt-and-pepper noise detection and reduction using fuzzy switching median filter”, *IEEE*, vol. 54, no. 4, pp. 1956-1961, 2008. [CrossRef] [Google Scholar] [Publisher Link]

[3] S.N. Sulaiman and N.A.M. Isa, “Denoising-based clustering algorithms for segmentation of low-level salt-and-pepper noise-corrupted images”, *IEEE*, vol. 56, no. 4, pp. 2702-2710, 2010. [CrossRef] [Google Scholar] [Publisher Link]

[4] S. Akkoul, R. Ledee, R. Leconge and R. Harba, “A New Adaptive Switching Median Filter”, IEEE, vol. 17, no. 6, pp. 587-590. [CrossRef] [Google Scholar] [Publisher Link]

[5] M. Storath and A. Weinmann, “Fast Median Filtering for Phase or Orientation Data”, *IEEE*, vol. 40, no. 3, pp. 639-652, 2018[CrossRef] [Google Scholar] [Publisher Link]

[6] F. Taherkhani and M. Jamzad, “Restoring highly corrupted images by impulse noise using radial basis functions interpolation”, *IET image processing*, vol. 12, no. 1, pp. 20–30, 2017 [CrossRef] [Google Scholar] [Publisher Link]

[7] M. Zhang Y. Liu, G. Li, B. Qin and Q. Liu, “Advances Iterative scheme inspired network for impulse noise removal”, SPRINGER, *pattern analysis and applications*, vol. 23, no. 1, pp. 135-145, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[8] G. Yuan and B. Ghanem, “l0TV: a sparse optimization method for impulse noise image restoration”, *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 2, pp. 352-364, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[9] J. Chen, Y. Zhan, and H. Cao, "Adaptive sequentially weighted median filter for image highly corrupted by impulse noise", *IEEE access*, vol. 7, pp. 158545–158556, 2019[CrossRef] [Google Scholar] [Publisher Link]

[10] L. Jin, W. Zhang, G. Ma and E. Song, "Learning deep CNNs for impulse noise removal in images", *ELSEVIER, journal of visual communication and image representation*, vol. 62, pp. 193-205, 2019[CrossRef] [Google Scholar] [Publisher Link]

[11] Sonali, S. Sahu, A.K. Singh, S. Ghrera, and M. Elhoseny, "An approach for de-noising and contrast enhancement of retinal fundus image using CLAHE", *ELSEVIER, optics & laser technology*, vol. 110, pp. 87-98, 2019[CrossRef] [Google Scholar] [Publisher Link]

[12] X. Zhang, J. Zheng, D. Wang, and L. Zhao, 2020, "Exemplar-based denoising: a unified low-rank recovery framework", *IEEE transactions on circuits and systems for video technology*, vol. 30, no. 8, pp. 2538-2549. [CrossRef] [Google Scholar] [Publisher Link]

[13] C.J.J. Sheela and G. Suganthi, "An efficient de-noising of impulse noise from MRI using adaptive switching modified decision based un-symmetric trimmed median filter", *ELSEVIER, biomedical signal processing and control*, vol. 55, pp. 1-12, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[14] G. Li, X. Xu, M. Zhang, and Q. Liu, "Densely connected network for impulse noise removal", *SPRINGER, pattern analysis and applications*, vol. 23, issue 3, pp. 1263–1275, 2020[CrossRef] [Google Scholar] [Publisher Link]

[15] Q. Wang, J. Ma, S. Yu, and L. Tan, "Noise detection and image de-noising based on fractional calculus", *ELSEVIER, chaos, solitons & fractals*, vol. 131, 2020[CrossRef] [Google Scholar] [Publisher Link]

[16] Q. Liu and Y. He, "Robust geman-mcclure based nonlinear spline adaptive filter against impulsive noise", *IEEE access*, vol. 8, pp. 22571–22580, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[17] G.M. Mukti, Maniruzzaman M.A. Alahe, A. Sarka, "Noise Removal from MRI Brain Images Using Median Filtering Techniques", *IJRES*, vol.10, no. 6, pp. 736-743, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[18] X. Yan, M. X. Xiao, W. Wang, Y. Li, F. Zhang, "A Self-Guided Deep Learning Technique for MRI Image Noise Reduction", *JTPES*, vol. 4, no. 1, 2024. [CrossRef] [Google Scholar] [Publisher Link]

AUTHORS

S. Prathiba She received the B.Sc. degree in Mathematics from Meenakshi College for Women (Autonomous), Chennai, in 2008 and the M.Sc. degree in Information Technology from Meenakshi College for Women (Autonomous), Chennai, in 2010. She received the M.Phil. degree in Computer Science from S.T. Hindu College, Nagercoil, in 2019. She is currently pursuing the Ph.D. degree in Computer Science at Manonmaniam Sundaranar University, Tirunelveli. Her research interest includes digital image processing and business intelligence.

B. Sivagami She received the B.Sc. degree in Computer Science from S.T. Hindu College, Nagercoil, in 1991 and the M.Sc. degree in Computer Science from Avinashilingam Deemed University, Coimbatore, in 2006. She received the M.Phil. degree in Computer Science from Azhagappa University, Karaikudi, in 2001 and the Ph.D. degree in Manonmaniam Sundaranar University, Tirunelveli, in 2014. She joined the Department of Computer Science & Application, S.T. Hindu College in 1993 as an Assistant Professor. Since 2019, she has been working as Associate Professor and Head of the Computer Science & Application department in S.T. Hindu College. Her research interest includes digital image processing.

Arrived: 10.07.2024
Accepted: 14.08.2024

International Journal of Data Science and Artificial Intelligence (IJDSAI)
Volume 02, Issue 04, July – August (2024)

RESEARCH ARTICLE

CHICKEN SWARM OPTIMIZATION BASED ENSEMBLED LEARNING CLASSIFIER FOR BLACK HOLE ATTACK IN WIRELESS SENSOR NETWORK

K. Vijayan[1, *], S.V. Harish[2] and R.A. Mabel Rose[3]

[1]Department of Electronics and Communication Engineering, Sapthagiri NPS university, Banglore, Karnataka 560057, India.
[2]Department of Electronics and Communication Engineering, NIE(South), Mysuru, Karnataka 570008, VTU Affiliated College, India.
[3]Department of Computer science and Engineering, Panimalar Engineering College, Varadharajapuram, Poonamallee, Chennai, 600 123, India.

[*]*Corresponding e-mail: vijayankvijayan@gmail.com*

Abstract – Wireless Sensor Networks (WSNs) are an inevitable technology prevalently used in various critical and remote monitoring applications. The security of WSNs is compromised by various attacks in wireless medium. Even though, various attacks are present, the Black hole attack degrades the network performance and resource utilization resulting in poor network lifetime. Therefore, the proposed research suggests an effective Intrusion Detection System for WSN to detect and classify black hole attacks based on ensemble ML classifiers. The BDD dataset is used for the analysis which is subjected to Chicken Swarm Optimization based feature selection. The selected features are balanced through SMOTE and TOMEK based STL data balancing module. An ensemble of five baseline ML classifiers such as SMO, NB, J48, KNN and RF utilizing voting ensemble approach is suggested to classify the attacks in the dataset. The performance of the algorithm is analyzed through evaluation metrics such as accuracy, precision, recall and F1-score. The comparison of proposed model with six ML and DL classifiers exposes the superiority of the proposed model's classification performance.

Keywords *– WSN, Black hole attack, SMOTE, BDD dataset, Chicken Swarm Optimization, Ensemble classifier.*

1. INTRODUCTION

Since WSNs function in a limited resources environment, they differ from the typical OSI paradigm. WSN sensor nodes are grouped or clustered closely around a certain region which is defined as sensor area. These nodes are maintained remotely and have limited computational power and bandwidth. Because nodes in WSNs are frequently left unattended, an attacker can simply seize a node. Furthermore, the nodes such as sensors in WSN are susceptible for a variety of malfunctions, and the communication link is unstable. As a result, WSN security is both a tough and critical task [1]. In general, assaults are classified as active or passive. Active assaults include black holes, wormholes, flooding, and overlay network wormholes [2]. Among them, black hole attacks can significantly affect network performance and resources [3, 24]. Several approaches for detecting and preventing such assaults have been developed. A targeted feature selection technique for determining the most significant traits may be advantageous [23]. The selected characteristics can then be implemented for designing an effective learning model for classification. Moreover, the adoption of attack-specific dataset can aid in enhanced speed and accuracy of detection [25].

In the proposed study, a metaheuristic optimization-based Chicken Swarm Optimization features selection, dataset balancing and voting ensemble ML classifier are used to design a Black hole attack classification-based IDS in WSN. The suggested algorithm's classification success has been evaluated with metrics such as accuracy, recall, confusion matrix, precision and F1-score on the BDD dataset. The following are the primary contributions of the proposed research:

- A Black hole attack detection system specialized to WSNs based on classification was built in the study. In identifying WSN-specific intrusions, a voting ensemble ML strategy including Random Forest, KNN, SMO, Naive Bayes, and J48 as base classifiers.
- On the BDD dataset, feature selection was conducted using Chicken Swarm Optimization (CSO) to reduce computational complexity while increasing classification accuracy.
- For data balance, the STL link that contains oversampling with SMOTE and under sampling with Tomek-Links methods are coupled. The drawbacks of both oversampling and under sampling approaches are avoided, resulting in improved classification performance.

Section 2 provides the review of literature relevant to black hole detection, Section 3 explains the behavior of BH attack, Section 4 explains the proposed methodology and Section 5 provides the simulation and experimentation results of the study. Section 6 describes the conclusions of the study.

2. RELATED WORK

A DL based WSN black hole and wormhole attack detection framework has been designed by Pawar, M. V. in 2023. The attack classification has been performed by LSTM and Whale Optimization Algorithm based on Fitness Rate (FR-WOA) can calculate the shortest path along with Round Trip Time (RTT) validation process and Bait process [4]. To decrease BH assaults, Dhanaraj, R. K., et al. proposed an Enhanced Gravitational Search Algorithm (EGSA) module for Simulated Annealing Black-hole assault Detection (SABD) in 2021. EGSA-SABD is used to detect and isolate attacking nodes in WSN [5].

H. Kalkha suggested a Hidden Markov Model approach to recognize fraudulent nodes in WSNs by preventing black hole attacks in 2019. It proposed a novel routing method that assesses the shortest way to prevent malicious node paths [6]. Suma, S., and B. Harsoor (2022) used On-demand Link and Energy Aware Dynamic Multipath (O-LEADM) routing strategy for MANETs to identify black-hole node by incorporating bait approach to discriminate packet loss due to congestion or malicious node. While accessing the channel, the activity of the node is analyzed employing control messages reply-sequence (rep-Seq) and destination-sequence (des-Seq) [7]. Gite, P., et al. 2023 proposes a lightweight model for identifying black hole, wormhole, grey hole, and DDoS attacker nodes in a WSN with no sensor node burden and uses the C4.5 and CART classifiers (decision tree algorithms) [8]. J. Kolangiappan and A. S. Kumar (2022) proposed a blackhole attack avoidance strategy based on a Deep Belief Network (DBN) with a larger number of hidden layers. [9].

Umamaheswari, S., et al. develop an IDS to categorize WSN assaults on WSN-DS dataset using ML classifier in 2021 to assess system performance. For feature extraction, a decision tree classifier is employed, and feature selection is accomplished using Fisher Score, Correlation Score, and Kruskal-Wallis (KW) based Statistical Analysis, Relief algorithm and Minimum Redundancy Maximum Relevance (MRMR) method [10]. Pawar, M. V., and A. Jagadeesan (2021) proposed IDS for WSN that detects blackhole and wormhole threats. A Self Adaptive-Multi-Verse Optimization (SA-MVO) approach is used to extract the optimum unique characteristics. Subsequently, the best characteristics were exposed to Deep Belief Network (DBN) analysis [11].

In 2022, Tabbaa, H., et al. investigate the use of different homogeneous ensemble HAT and an Adaptive Random Forest (ARF) based heterogeneous ensemble paired to the Hoeffding Adaptive Tree (HAT) algorithm in WSN-DS dataset to recognize attack types: Grayhole, Blackhole attack, Scheduling and Flooding across WSN traffic [12]. Rezvi, M. A., and colleagues will present in 2021 a data mining approach for different kinds of classification algorithms to identify Grayhole, Blackhole, Flooding, and TDMA. Several data mining approaches, including Support Vector Machine (SVM), KNN, Nave Bayes, Logistic Regression, and ANN algorithms, are used to the dataset and their performance in identifying assaults is evaluated [13].

3. BLACK HOLE ATTACKS IN WSN

If a suspicious node enters a network and captures the transferred data traffic along the network and drops the data packets without further transmission is termed as black hole attack and it produces Denial of Service (DoS) [14]. The attacker selectively drops packets or all control and data packets routed through him in this attack. This is done in two phases by malicious nodes. First, the attacker node displays a fictitious low-rank value in order to entice neighbors to choose it as their parent (Sink hole attack) [15]. Second, it may drop selected packets depending on predetermined criteria (Selective Forwarding attack), or it might drop all packets from other nodes [16]. As a result, every packet passing via this intermediary malicious node is susceptible to partial or complete data loss. Black hole attacks have an impact on network performance. Black hole attacks have a large impact on throughput, packet delivery rate, and latency, but only a moderate impact on battery drain and control packet overhead [17].

In general, there are two ways for the malicious node to obtain the data packet in this type of attack. In first method, a Route Reply control message (RREP) message is send using the routing protocol to the source node by the malicious node as soon as a receiving a Route Request control message (RREQ) is received in order to enter the network as neighboring node having shortest path to reach destination. Such bogus route can be used by the source node to transfer data packets. Second, whether the malicious node is capable of intercepting data transfers without transmitting control message (RREP) to source node. Under both cases, the malicious node dumps the transferred data on receiving it. Consequently, the data transfer from source to destination is interrupted affecting connection and performance of network.

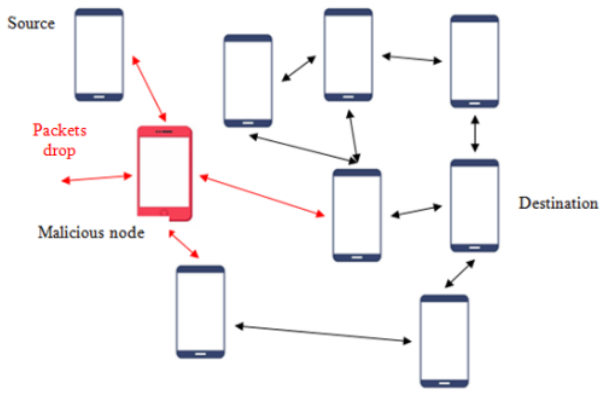


Figure 1. Illustration of black hole attack

Figure 1 illustrates the black hole attack scenario. "Source" node intends to communicate with node "Destination," thus it broadcasts "RReq" to the neighboring nodes. Node "M" injects itself and responds quickly,

claiming to have the best path. When communication begins, node "M" discards any data supplied through it.

Algorithm for Black hole Attack

Necessity: Node-ID of Attacker

If ID of Node is similar to ID of Attacker, then

Reduce the rank value

Retain higher rank parents

Discard data packets generated by nodes other than parents

else

Maintain clear calculations of rank

End if

4. PROPOSED METHODOLOGY

The proposed blackhole attack classification using ensemble ML classifier, CSO feature selection and STL data balancing approach consists of four phases. Figure 2 illustrates the representation of proposed method.

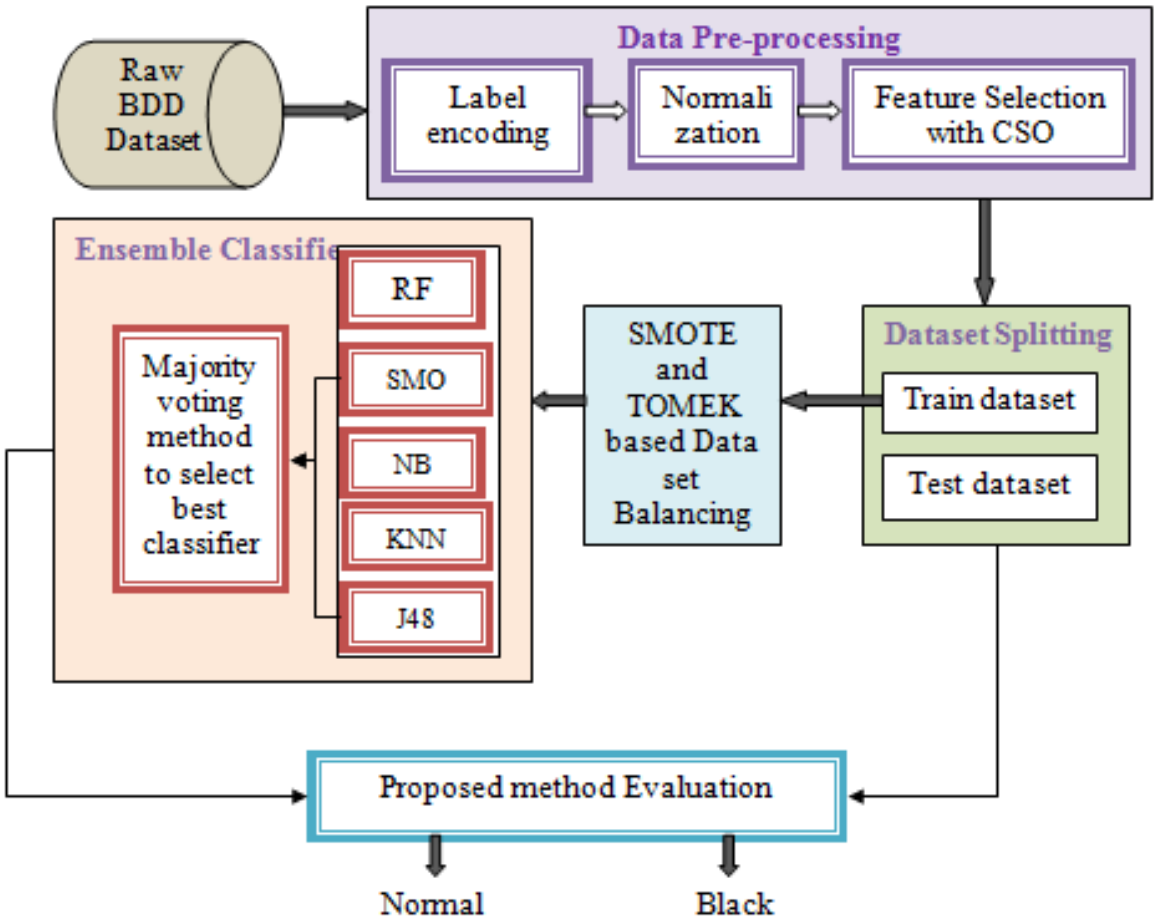


Figure 2. Overview of Proposed method

The initial phase is data preprocessing of BDD dataset. The data preprocessing consists of label encoding with one-hot encoding, normalization and feature selection. In this work, the Chicken Swarm Optimization (CSO) method was utilized for selecting the ideal feature combination to maximize classification performance while minimizing the number of chosen features. The next phase is data splitting into training and testing dataset. The third phase is data balancing to mitigate the imbalance problems in dataset. The final phase is classification based on ensemble ML classification with voting ensemble approach.

4.1. Dataset

The behavior of MANETs in the presence of black hole assaults is investigated in order to build the Behavior-Driven Development (BDD) dataset. This is designed especially for black hole attacks with labels. [18]. GloMoSim 2.03 simulator was utilized to collect audit data by simulating regular and black hole attack situations. There are 29 characteristics in the BDD dataset. Moreover, there are 1289 instances/nodes in dataset having 1189 normal nodes and 100 black nodes. Figure 3 depicts the characteristics observed in the BDD dataset.

4.2. Normalization and Label Encoding

The raw BDD dataset is subjected to One-hot encoding process to convert the categorical values as numerical values before proceeding to classification algorithm. The mathematical representation of normalization is explained in equation 1. It converts the numerical entries in the dataset into a value among 0 and 1.

$$x^n = \frac{x-\overline{x}}{\emptyset} \quad (1)$$

where x represents the actual entry, x^n represents the normalised value, $\overline{x}$ represents the mean and $\emptyset$ represent the standard deviation, accordingly. This normalization process reduces entries having high numerical value without affecting the classification output and network performance. This one hot encoding procedure alters labels/classes as value.

Total RREQ transmitted	Total RREP transmitted	Total RREP forwarded	RREP transmitted with route	Average Nodal speed
Total RERR transmitted	Total RERR retransmitted	Total selected routes	Total transmitted data	Total generated data
Total received data	Total discarded data	Total hop counts	Total control packets transmitted	Total retries of broken link
Total number of broken links	Total received RREQ	Total received RREP	Total data packets in node	Total low hop counts to destination
Total high hop counts to destination	Intermediate hops count	No. of Nodes transmitting maximum data	Highest number of reply from node	Total number of bytes transmitted
Total number of bytes received	Nodes functioning as source	Nodes functioning as destination	Fast replying AVG	

Figure 3. Features in BDD Dataset

4.3. Feature Selection

The Features Selection Algorithm may be used to locate the most important and trustworthy characteristics in finding and preventing black hole attacks before incorporating the entire BDD dataset into the classifier. Using the feature selection procedure resulted in a reduction in model complexity and computation time, which helped in designing the optimal learning model.

CSO Algorithm

CSO algorithm has three classifications of roles: roosters, hens, and chicks, each with its own set of behavior criteria. The following are the fundamental preliminaries of CSO algorithm.

i. A chicken swarm is divided into groups each having a rooster, few chickens and more hens.

ii. Roosters, chickens and hens have unique characteristics mentioned by corresponding fitness score. Based on which, the chicks are worst, roosters are best and hens are inter mediate. Each hen picks one rooster at random as her mate and joins his group, and each chick chooses one hen at random as its mother.

iii. For iterative cycle of G generations, the unique identities, maternal and spouse relationships remain unaltered. After G generations all these values are updated.

iv. Hens follow their partner rooster to locate food in each group of the entire population and thrive to get food at random among other members in a group. Members with high fitness can get food.

The position of each chicken describes its location. Let $R_n,\ C_n,\ H_n, and\ M_n$ denote the number of rooster, chicken, hen and mother hens correspondingly. Let the position of i^{th} chicken in the t^{th} on the j^{th} dimensional space be $x_{i,j}^t$. The value of t lies between$\{1,2, \dots, T\}$, j lies between $\{1,2,3, \dots J\}$ and i lies between $\{1,2, \dots I\}$ in which the number of chickens is represented by I, the dimension number is represented by J and the maximum iterations are denoted by T. Individual location update formula exists for a chicken, a hen and a rooster. The rooster's recurrent location can be mathematically represented as,

$$x_{i,j}^{t+1} = (Randn(0, \emptyset^2) + 1) \quad (2)$$

$$\emptyset^2 = \begin{cases} 1, & if\ f_i \le f_k \\ \exp\left(\frac{f_k - f_i}{\varepsilon + |f_i|}\right) & otherwise, k \in [1, NR], where\ k \neq i \end{cases} \quad (3)$$

In the above equation, $(Randn(0, \emptyset^2)$ is the random number having $\emptyset^2$ variance and zero expectation following Gaussian distribution. k denotes the number of randomly selected rooster, ε denotes a constant of small value, the fitness value of k^{th} and i^{th} rooster is f_k and f_i accordingly. The hen's recurrent location can be denoted mathematically as,

$$x_{i,j}^{t+1} = Rand * C_1 * (x_{r1,j}^t - x_{i,j}^t) + Rand * C_2 * (x_{r2,j}^t - x_{2,j}^t) x_{i,j}^t \quad (4)$$

The random number $Rand$ lies between [0,1] in the uniform distribution, the learning factors denoted by C_1 and C_2. The hen's recurrent location is mathematically represented as,

$$x_{i,j}^{t+1} = RF * \left(x_{m,j}^t - x_{i,j}^t\right) + x_{i,j}^t \quad (5)$$

The term RF denotes the random factor in the range [0,2], $x_{m,j}^{t}$ denotes the mother hen.

Input: parameters N, T, NR, NH, NC, NM, G

Output: Selection of best features

Steps:

(1) Randomly assign locations to chickens.

(2) Each chickens' fitness score is calculated; the local and global optimal location of each chicken is selected. Iteration time is set to $t = 1$.

(3) If $t\%G = 0$ (% is the remainder function), then based on descending order of fitness score, chickens are arranged. Roosters, NR have best values, NC chicks have worst value and NH hens are others. The swarm is classified as groups having a rooster, various hens and chicks in which moms and spouses are randomly selected.

(4) The positions of roosters, chicks and hens are updated using formulae (2), (4), and (5), and fitness values are evaluated.

(5) The population's global best location and each individual's local best position are updated.

(6) Iterate $t = t + 1$ times; if t equals M or the solution meets the accuracy criteria, CSO outputs the final result; otherwise, go to Step 3.

CSO based Feature Selection

The following four characteristics are identified as the most significant features based on the analysis results: (1) Total RREQ features transmitted. (2) Total RREPs with forward feature. (3) High destination sequence number features. (4) A low number of hops to the destination feature. Furthermore, despite the fact that they do not have a very significant fitness value, the findings of the evaluation of these two features indicate that they're capable of an important impact in prevention and detection of BH attack, and they are: (5) The total acts that serve as the source feature. (6) Act as a destination feature. The BDD dataset after CSO based feature selection has six features alone that are most significant for classification.

4.4. Dataset Splitting

During the dataset splitting step, the BDD dataset is separated into a training dataset and a testing dataset. The training step employs a labeled training set to train a particular classifier, which is then utilized in the testing phase to categorize test instances as black or normal.

4.5. Data Balancing

The unequal distribution of classes has a detrimental impact on categorization performance. Minority groups, in particular, have a detrimental impact on the detection rate. IDS designs cannot adequately identify the class imbalance issue in dataset. The individual use of undersampling approach for class imbalance leads to removal of important data transmitted and reduces data quality significantly. The use of oversampling approach results in unwanted noise and data volume increase. To address the unbalanced class problem, Synthetic Minority Oversampling Technique (SMOTE) oversampling and Tomek-Links undersampling approaches named as STL approach is presented in this work.

SMOTE

SMOTE was suggested by Chawla et al. [19], a heuristic oversampling approach, to overcome the class imbalance issue in datasets. Minority class data are oversampled to generate synthetic data in this approach. The overfitting issue is also eliminated with the generated synthetic data. The overfitting problem of class imbalance can be reduced with random non-metaheuristic sampling approach which is widely used recently [20]. The class imbalance issue is eliminated with SMOTE through increasing the number of minorities labeled instances along with its neighbors. Samples in proximity to the feature space are selected by SMOTE. An arbitrary instance is selected from the minority label and its proximal neighbor's k is identified. A neighbor is selected at random and the variation among both samples is multiplied with a value from 0 to 1 which is combined with randomly selected sample value. The line created across the two sample attributes is then used to create synthetic samples. Randomly selected neighbors from k proximal neighbors determine the required oversampling quantity.

The linear groupings of two minority labeled identical samples $(X, X^r)$are mathematically represented as,

$$m = x + g.(X^r - X), \quad 0 \le g \le 1 \tag{6}$$

The sample X was selected randomly as X^rin correspondence to the nearest proximal number having difference g among two instances.

TOMEK-LINKS

Tomek-Links is a Tomek-developed approach for undersampling unbalanced datasets. It may be thought of as an enhanced variant of the Nearest Neighbour Rule. The specimens on the Tomek link can be deleted from the provided dataset using this method [21]. It generates sample data pairs within the same dataset and from separate labels. These paired data are referred to as Tomek linkages [22]. Its primary aim was to segregate the majrotiy and minority labels. Let u, v be the proximal neighbors and u belong to a class and v belongs to another class. The distance among instances u and v is represented as,

$$Tomex\ link\ be\ T(u,v), for\ any\ value\ of\ i, * d(u,v) < d(u,i) ord(u,v) < d(v,i) \tag{7}$$

T-links connect the two classes. This link's data samples are deemed noise. The removal of majority class noises improves class separation and stabilizes the data distribution. Thus, the noise instances are eliminated from the majority labels.

4.6. Ensemble classifier to detect Black hole attack

This module compares the test data to the network's typical profile using a predetermined classifier to determine if the data is normal or malicious. If there is any divergence from the network's typical behavior, the incident is classified as an attack. Otherwise, it is seen as normal. Classification is

the process of learning a model (classifier) from a training module with labeled data to categorize test data as labels. Ensemble learning is a popular ML approach employed for the categorization and detection of WSN-attacks. These strategies are often built by solving the identical problems with different ML classifiers and combining the results with one of the voting procedures. It combines many basic models to create one optimal prediction model. Bagging, boosting, stacking, and majority voting are examples of ensemble procedures. We describe an ensemble-based ML technique for detecting WSN-attacks by combining a number of different base models. To distinguish the most effective classifier for detecting assaults, many classifiers have been trained to test the evaluation measures. This suggested method was developed utilizing a collection of heterogonous supervised ML approaches, including SMO, NB, J48, KNN, and RF. Ensemble methods are commonly used in machine learning approaches.

Sequential Minimal Optimization (SMO) is a novel training approach for Support Vector Machines (SVM). SMO was created in order to address the Quadratic Programming (QP) SVM training issue. A Naive Bays (NB) algorithm is a basic probabilistic algorithm for classification which uses the Bayes' theorem to categorize a fresh occurrence based on robust independent assumptions about the characteristics. J48 classifier was a basic incorporation of the C4.5 decision tree technique utilizing the training set's attribute values to generate a binary tree. A non-parametric K-Nearest Neighbor (K-NN) supervised technique for regression and classification issues that may quickly determine the category or class of a given dataset. Random Forest algorithm is a classification process on dataset using various decision trees on various data sub-groups. The prediction accuracy of provided dataset can be enhanced through averaging them. In this study, the majority voting approach is used to combine predictions from many other models. When employing majority voting, performance can be improved over when using a single model. When all models perform equally well, the voting ensemble approach is utilised. As basic models, SMO, NB, J48, KNN, and RF are employed. Different classifiers combine using the majority vote approach in majority voting. The greatest likelihood of the chosen class determined the final forecast. Figure 4 depicts the ensemble voting mechanism.

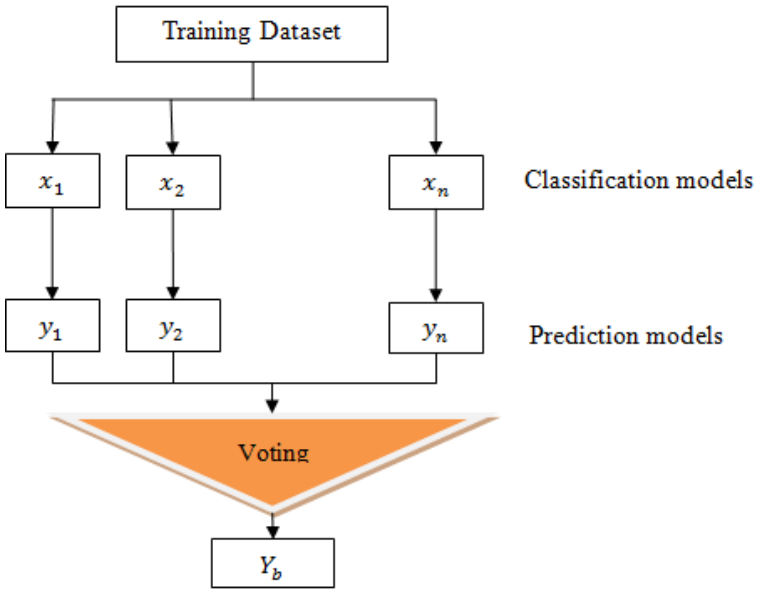


Figure 4. Voting based Ensemble method

Individual learners SMO, NB, J48, KNN, and RF, as well as majority voting, are implemented on the BDD dataset utilizing feature selection and data balance in the suggested technique. The results of both methods are then compared using the criteria accuracy, precision, and recall.

5. SIMULATION RESULTS

The suggested approach has been evaluated on the unbalanced BDD dataset in this part. The research is carried out in Python programming language with Pyspark tool, using Google Colab framework in Apache Spark environment. Keras and Scikit-learn libraries of PySpark MLib tool has been employed for ensemble, ML and DL techniques. The suggested technique was tested against six various ML and DL algorithms, and the results were analyzed.

5.1. Evaluation Metrics

The study uses the famous metrics such as accuracy, precision, recall and F1-sore. Many classification problems make use of these assessment factors. The confusion matrix data is used to calculate these parameters. The confusion matrix's primary constituents are true-positive (t_p), false-positive (f_p), true-negative (t_n), and false-negative (f_n). Accuracy defines the percentage of rightly identified samples. Precision can be termed as the ratio of the number of retrieved relevant instances to the number of retrieved samples (relevancy and non-relevancy). Recall is the ratio of rightly categorized nodes as black to the total black nodes in the data set. A harmonic mean of Recall and Precision is F-score. The mathematical formula for accuracy, precision, recall and F-score is explained below in equations.

$$Accuracy = \frac{t_p+t_n}{t_p+f_p+t_n+f_n} \quad (8)$$

$$Recall = \frac{t_p}{f_n+t_p} \quad (9)$$

$$Precision = \frac{t_p}{t_p+f_p} \quad (10)$$

$$f1-score = \frac{2t_p}{\frac{t_p}{2t_p+f_n+f_p}} \quad (11)$$

5.2. Performance Analysis

The performance of the proposed ensemble ML classifier has been examined based on the above evaluation metrics. The values of accuracy, precision, recall and F1-score for the baseline ML models such as SMO, NB, J48, RF and Voting Ensemble classifier are calculated and its numerical values are illustrated in Table 1.

Table 1. Performance Evaluation of proposed model

Models	Accuracy	Recall	Precision	F1-score
SMO	0.9876	0.9958	0.9907	0.9932
NB	0.9930	0.9983	0.9941	0.9962
J48	0.9961	0.9992	0.9966	0.9979
KNN	0.9899	0.9924	0.9966	0.9945
RF	0.9977	0.9992	0.9983	0.9987
Ensemble Classifier with Majority Voting	**0.9953**	**0.9983**	**0.9966**	**0.9975**

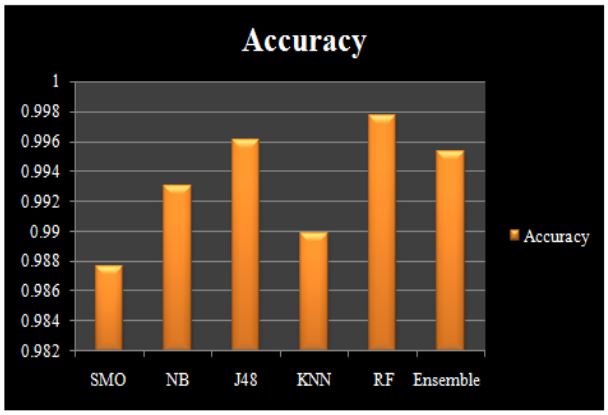


Figure 5. Accuracy Comparison of baseline models and ensemble classifier

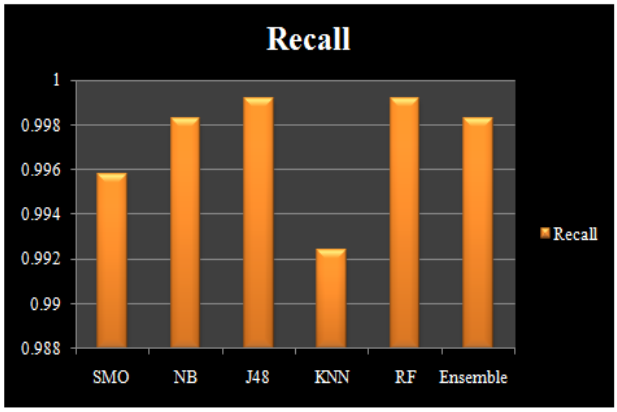


Figure 6. Recall Comparison of baseline models and ensemble classifier

As depicted in Table 1, the RF classifier has the highest accuracy of 0.9977, recall of 0.9992, precision of 0.9983 and F1-value of 0.9987 among other baseline classifiers. The ensemble classifier produces values of 0.9953 accuracy, 0.9983 recall, 0.9966 precision and 0.9975 F1-score respectively. The graphical representation of comparison of accuracy of five base models and proposed model is illustrated in Figure 5. Similarly Figure 6, 7 and 8 represent the graphical comparison of recall, precision and F1-score values respectively.

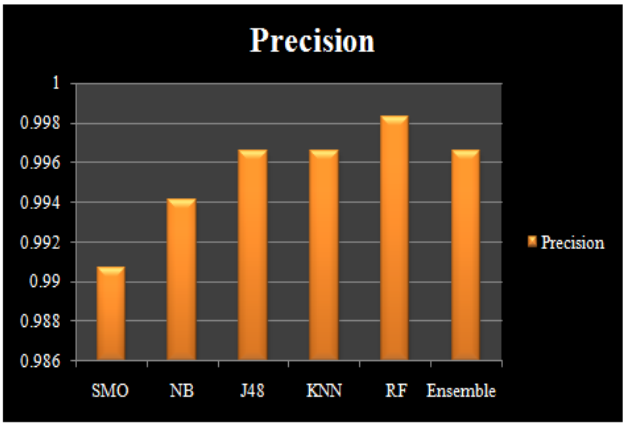


Figure 7. Precision Comparison of baseline models and ensemble classifier

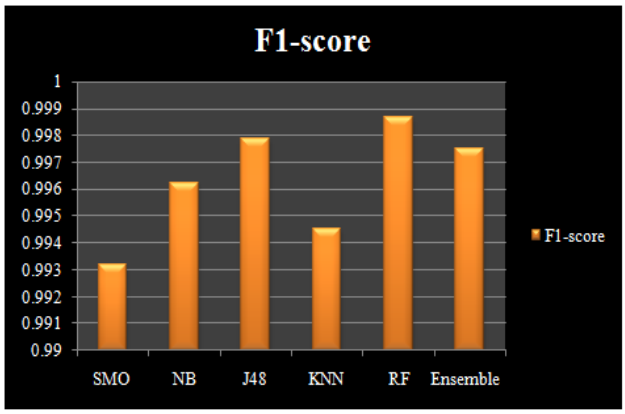


Figure 8. F1-score Comparison of baseline models and ensemble classifier

Table 2. Confusion matrix for classification baseline models and ensemble classifier

	RF		KNN		NB		J48		SMO		Ensemble	
Actual vs. Predicted	**Normal**	**Black**	**Normal**	**Black**	**Normal**	**Black**	**Normal**	**Black**	**Normal**	**Black**	**Normal**	**Black**
Normal	1184	2	1178	4	1181	7	1185	4	1177	11	1183	4
Black	1	100	9	98	2	99	1	99	5	96	2	100

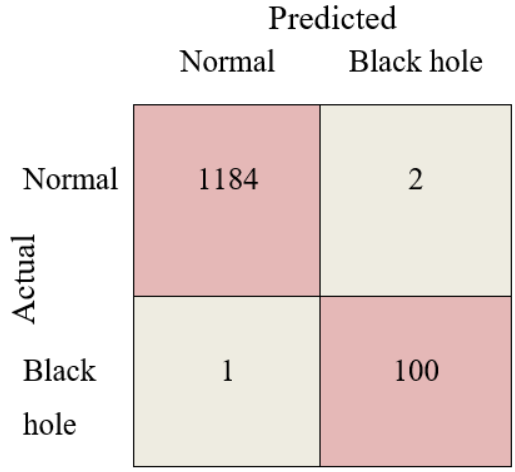


Figure 9. Confusion matrix of RF classifier

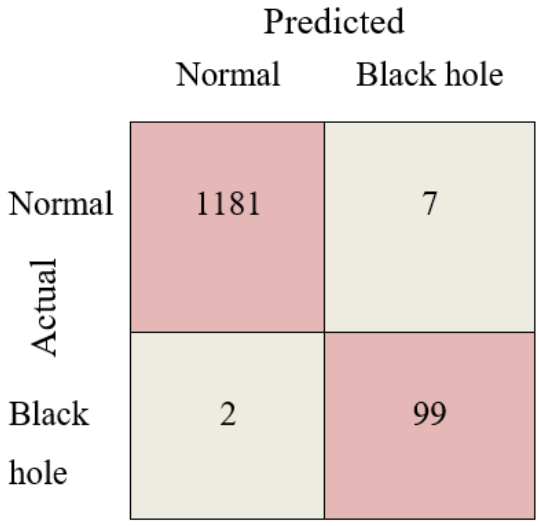


Figure 10. Confusion matrix of NB classifier

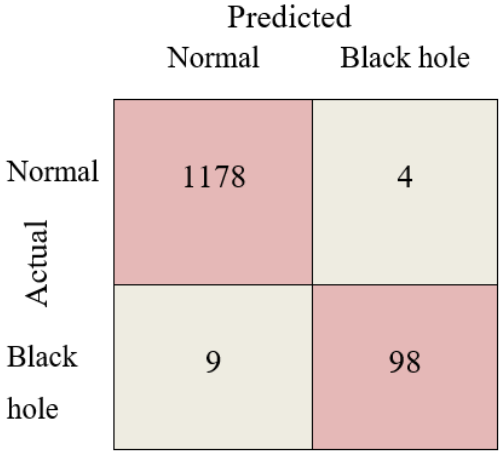


Figure 11. Confusion matrix of KNN classifier

The value of actual classes to the predicted classes are explained in Table 2 and from its interpretation, the RF classifier has the best classification performance among the comparable baseline models.

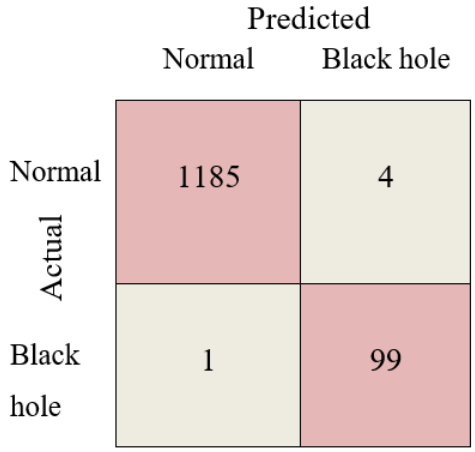


Figure 12. Confusion matrix of J48 classifier

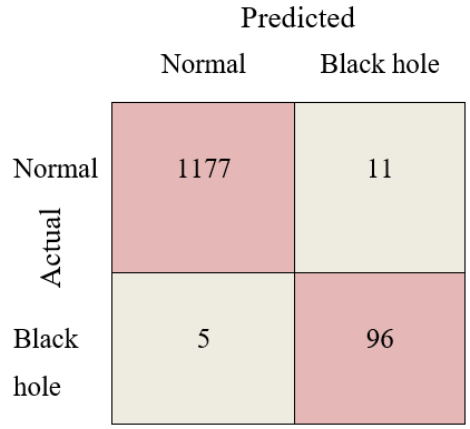


Figure 13. Confusion matrix for SMO classifier

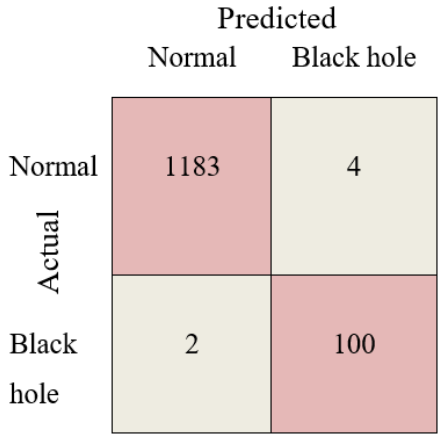


Figure 15. Confusion matrix for Ensemble classifier

5.3. Comparative Analysis

The effective performance of the proposed model can be analyzed through evaluation with other similar existing models. In the study, the accuracy, precision, recall and F1-score values are compared with three ML methods and three DL methods in order to examine the proposed model's classification performance.

Table 3. Comparison of Performance of various models with proposed ensemble classifier

Models	Accuracy	Recall	Precision	F1-score
SVM	0.9818	0.9965	0.9845	0.9905
Logistic Regression	0.9672	0.7181	0.9099	0.8028
ANN	0.9856	0.9124	0.9066	0.90954
CNN	0.9879	0.9297	0.9486	0.9372
DNN	0.9704	0.8201	0.8280	0.8208
RNN	0.9648	0.6911	0.8562	0.7537
Proposed	0.9953	0.9983	0.9966	0.9975

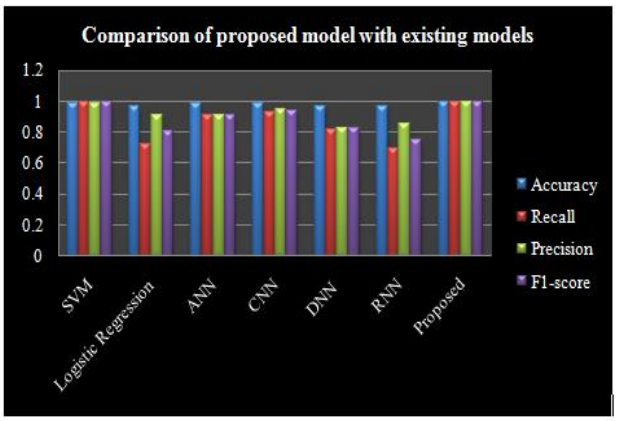


Figure 16. Comparison of proposed model with existing approaches

Table 3 provides the values of performance metrics of ML models such as Support Vector Machine (SVM) [13], Logistic Regression and Artificial Neural Network (ANN) [14] along with three DL models such as Deep Neural Network (DNN), Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) [15]. The graphical illustration of the performance comparison of the above models with proposed ensemble classifier is explained in Figure 16. From the results, the proposed model shows the highest accuracy, precision, recall and F1-score values of 0.9953, 0.9983, 0.9966 and 0.9975 respectively. Apart from this, the CNN model has the highest classification accuracy of 0.9879 among DL models and ANN model exhibit 0.9856 accuracy among ML models. In terms of recall, precision and F1-score, the SVM has highest value of 0.9965, 0.9845 and 0.9905 respectively followed by CNN with0.9297 recall, 0.9486 precision and 0.9372 F1-score respectively.

6. CONCLUSIONS AND RECOMMENDATIONS

The study proposes a classification-based IDS in WSN to detect black hole attack. The BDD dataset is used for the analysis. The dataset is preprocessed with label encoding and normalization followed by feature selection. The optimal features suitable for classification can be viewed as an optimization problem and can be selected with the swarm-based Chicken Swarm Optimization (CSO) approach. The CSO algorithm reduces the BDD dataset with 29 features into 6 features most significant for attack classification. The dataset with selected features are spitted into training and testing dataset. The training dataset is fed into data balancing module which eliminates the class imbalance problem in dataset using SMOTE and TOMEK Link (STL) approach for upsampling and downsampling. The ensemble classifier with five base ML classifiers such as RF, KNN, J48, NB and SMO are utilized and combined with voting ensemble method. The performance of the proposed model has been analyzed with accuracy, precision, recall and F1-score. The comparative analysis of the ensemble classifier with three ML and three DL models illustrates the superiority of the proposed classification approach. In future, the proposed study can be enhanced with adapting fuzzy based feature selection techniques. Moreover, the attack classification approach can be extended to detect other attack types such as wormhole attack, grayhole attack and flooding.

CONFLICTS OF INTEREST

The authors declare that there is no conflict of interest.

FUNDING STATEMENT

Not applicable.

ACKNOWLEDGEMENTS

The author would like to express his heartfelt gratitude to the supervisor for his guidance and unwavering support during this research for his guidance and support.

REFERENCES

[1] U. D. Maiwada, A. A. Muazu and N. Noor, "The Security Paradigm That Strikes a Balance Between a Holistic Security Mechanism and The WSN's Resource Constraints", *East Asian Journal of Multidisciplinary Research*, vol. 1, no. 3, pp. 343-352, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[2] M. Keerthika and D. Shanmugapriya, "Wireless sensor networks: Active and passive attacks-vulnerabilities and countermeasures", *Global Transitions Proceedings*, vol. 2, no. 2, pp. 362-367, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[3] G. Farahani, "Black hole attack detection using K-nearest neighbor algorithm and reputation calculation in mobile ad hoc networks", *Security and Communication Networks*, vol. 2021, pp. 1-15, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[4] M. V. Pawar, "Detection and prevention of black-hole and wormhole attacks in wireless sensor network using optimized LSTM", *International Journal of Pervasive Computing and*

Communications, vol. 19, no. 1, pp. 124-153, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[5] R. K. Dhanaraj and R. H. Jhaveri, L. Krishnasamy, G. Srivastava, & P. K. R. Maddikunta, "Black-hole attack mitigation in medical sensor networks using the enhanced gravitational search algorithm", *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 29, no. Suppl 2, pp. 297-315, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[6] H. Kalkha, H. Satori and K. Satori, "Preventing black hole attack in wireless sensor network using HMM", *Procedia computer science*, vol. 148, pp. 552-561, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[7] S. Suma and B. Harsoor, "An approach to detect black hole attack for congestion control utilizing mobile nodes in wireless sensor network", *Materials Today: Proceedings*, vol. 56, pp. 2256-2260, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[8] P. Gite, K. Chouhan, K. M. Krishna, C. K. Nayak, M. Soni and A. Shrivastava, "ML Based Intrusion Detection Scheme for various types of attacks in a WSN using C4. 5 and CART classifiers", *Materials Today: Proceedings*, vol. 80, pp. 3769-3776, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[9] J. Kolangiappan and A. S. Kumar, "A novel framework for the prevention of black-hole in wireless sensors using hybrid convolution network", 2022. [CrossRef] [Google Scholar] [Publisher Link]

[10] S. Umamaheswari, K. H. Priya and D. Allinjoe, "Towards Building Robust Song and Location Detection System Using LBP Features", In 2021 *International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA)*, pp. 1-6, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[11] M. V. Pawar and A. Jagadeesan, "Detection of blackhole and wormhole attacks in WSN enabled by optimal feature selection using self-adaptive multi-verse optimiser with deep learning", *International Journal of Communication Networks and Distributed Systems*, vol. 26, no. 4, pp. 409-445, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[12] H. Tabbaa, S. Ifzarne and I. Hafidi, "An online ensemble learning model for detecting attacks in wireless sensor networks", arXiv preprint arXiv:2204.13814, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[13] M. A., Rezvi, S. Moontaha, K. A. Trisha, S. T. Cynthia and S. Ripon, "Data mining approach to analyzing intrusion detection of wireless sensor network", Indonesian J. Electric. Eng. Comput. Sci, vol. 21, no. 1, pp. 516-523, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[14] A. Kumar, V. Varadarajan, A. Kumar, P. Dadheech, S. S. Choudhary, V. A. Kumar and K. C. Veluvolu, "Black hole attack detection in vehicular ad-hoc network using secure AODV routing algorithm", *Microprocessors and Microsystems*, vol. 80, pp. 103352, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[15] P. P. Ioulianou, V. G. Vassilakis and S. F. Shahandashti, "A trust-based intrusion detection system for RPL networks: Detecting a combination of rank and blackhole attacks", *Journal of Cybersecurity and Privacy*, vol. 2, no. 1, pp. 124-153, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[16] S. Singh and H. S. Saini, "Learning-based security technique for selective forwarding attack in clustered WSN", *Wireless Personal Communications*, vol. 118, no. 1, pp. 789-814, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[17] G. Farahani, "Black hole attack detection using K-nearest neighbor algorithm and reputation calculation in mobile ad hoc networks", *Security and Communication Networks*, vol. 2021, pp. 1-15, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[18] Y. Khamayseh, M. B. Yassein and M. Abu-Jazoh, "Intelligent black hole detection in mobile AdHoc networks", *International Journal of Electrical and Computer Engineering*, vol. 9, no. 3, pp. 1968, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[19] N. V. Chawla, K. W. Bowyer, L. O. Hall and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique", *Journal of artificial intelligence research*, vol. 16, pp. 321-357, 2002. [CrossRef] [Google Scholar] [Publisher Link]

[20] X. Tan, S. Su, Z. Huang, X. Guo, Z. Zuo, X. Sun and L. Li, "Wireless sensor networks intrusion detection based on SMOTE and the random forest algorithm", *Sensors*, vol. 19, no. 1, pp. 203, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[21] R. M. Pereira, Y. M. Costa and C. N. Silla Jr, "MLTL: A multi-label approach for the Tomek Link undersampling algorithm", *Neurocomputing*, vol. 383, pp. 95-105, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[22] M. Kamaladevi, V. Venkataraman and K. R. Sekar, "Tomek link Undersampling with Stacked Ensemble classifier for Imbalanced data classification", *Annals of the Romanian Society for Cell Biology*, pp. 2182-2190, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[23] A. I. Al-issa, M. Al- Akhras, M. S. ALsahli and M. Alawairdhi, "Using machine learning to detect DoS attacks in wireless sensor networks", In 2019 *IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, pp. 107-112, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[24] M. A. Rezvi, S. Moontaha, K. A. Trisha, S. T. Cynthia and S. Ripon, "Data mining approach to analyzing intrusion detection of wireless sensor network", *Indonesian J. Electric. Eng. Comput. Sci*, vol. 21, no. 1, pp. 516-523, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[25] S. Salmi and L. Oughdir, "Performance evaluation of deep learning techniques for DoS attacks detection in wireless sensor network", *Journal of Big Data*, vol. 10, no. 1, pp. 1-25, 2023. [CrossRef] [Google Scholar] [Publisher Link]

AUTHORS

K. Vijayan completed his PhD from Vels Institute of Science, Technology and Advanced Studies (VISTAS) formerly known as VELS University. His research interest includes Wireless sensor networks, Internet of things, VLSI circuits and systems, Machine learning, routing and networking in wireless sensor networks. He has completed his Masters in engineering in VLSI design from college of engineering Guindy - Anna University in 2003. He has 21 years of teaching experience and currently working as Professor in Department of Electronics and communication Engineering Sapthagiri NPS University. He has supervised 15 post graduate students and 45 undergraduate students for their academic projects. He is having two international patents and two national patents. He has published more than 25 papers in international journals and 7 papers in international conferences. He is also a life member of ISTE, IETE, SESI, IAENG and IACSIT.

S.V. Harish received his Post Graduate in 2010 from VTU, Belagavi. He is a assistant professor in the dept of Electronics and communication engineering NIE(South), Mysuru. He has 26 years of experience in academics 11 years in research field. His area of research interest are Wireless sensor networks, communication protocols and power electronics.

R.A. Mabel Rose received her B.E. degree and M.E. degree in Computer Science and Engineering from Anna University, Chennai, India. She started her career as Lecturer and has 13 years of experience. Currently she is working as Assistant Professor in Panimalar Engineering College, Chennai. Her research interests include Cyber Security and Cloud Computing. She is a lifetime member of ISTE.

Arrived: 16.07.2024
Accepted: 21.08.2024

International Journal of Data Science and Artificial Intelligence (IJDSAI)
Volume 02, Issue 04, July – August (2024)

RESEARCH ARTICLE

IOT BASED AIR QUALITY MONITORING USING DENSENET IN URBAN AREAS

M. Devaki [1, *], Jeyaraman Sathiamoorthy[2] and M. Usha [3]

[1] Department of Electrical and Electronics Engineering, Velammal College of Engineering and Technology, Madurai India.

[2] Department of Computer science and Design, RMK Engineering College (Autonomous), RSM Nagar, Kavarapettai, Gummidipoondi Taulk, Thiruvallur, 601206, India.

[3] Department of Master of Computer Application, MEASI Institute of Information Technology, Royapettah, Chennai 600014, India.

[*]*Corresponding e-mail: mde@vcet.ac.in*

Abstract – Internet of Things is being used more and more in the control and monitoring of air quality. Real-time data regarding air pollutants and other environmental parameters can be gathered by deploying IoT devices with sensors and connectivity capabilities. Rapid urbanization and industry cause increasingly serious problems with air quality. A significant challenge in the current air quality monitoring system is its limited spatial coverage and accuracy. In this paper, a novel air quality monitoring using IoT is proposed to monitor the quality of the air efficiently in real time. Sensors are placed in the various traffic system to collect environmental data and processed it in Real Time Data Analytics Module (RTDM). DenseNet is used to predict the quality of air and classified into three classes namely pure, impure, and normal. The efficacy of the proposed technique has been evaluated using assessment actions such as accuracy, time efficiency, precision, F1 score, RMSE, MAPE, and MAE. By the comparison analysis, the proposed technique's accuracy rate is 10.08%, 17.64%, and 34.34% higher than the existing Ide Air, SMOTEDNN, and ETAPM-AIT techniques respectively.

Keywords *– Air pollution, DenseNet, Sensors, Internet of Things, Real-Time Data Analytics Module.*

1. INTRODUCTION

This Internet of Things (IoT) has transformed the field of air monitoring systems by bringing intelligent, networked technologies for gauging and analyzing air quality [1]. IoT-based air monitoring systems combine state-of-the-art sensor technology with wireless connectivity to allow real-time data gathering and transfer to cloud-based platforms [2]. These interconnected sensors may detect particles, nitrogen dioxide, ozone, carbon monoxide, and particulate matter, among other air contaminants [3]. The vast and continuous data collection capabilities of these sensors allow for precise and comprehensive indoor and outdoor air quality monitoring. The data is then analyzed using advanced analytics, providing significant insights into patterns and trends in pollution [4].

Air pollution has become a serious issue worldwide, particularly in emerging countries, due to the rapid rise of manufacturing and urbanization [5]. Dangerous levels of particle matter, carbon monoxide, nitrogen dioxide, sulfur dioxide, ground-level ozone, volatile organic compounds, and carbon monoxide are associated with an increase in air pollution [6]. Industrial emissions and vehicular emissions are the main cause of air pollution [7]. When companies grow, they emit a range of dangerous chemicals into the environment because they need fossil fuels for transportation, manufacturing, and electricity production [8].

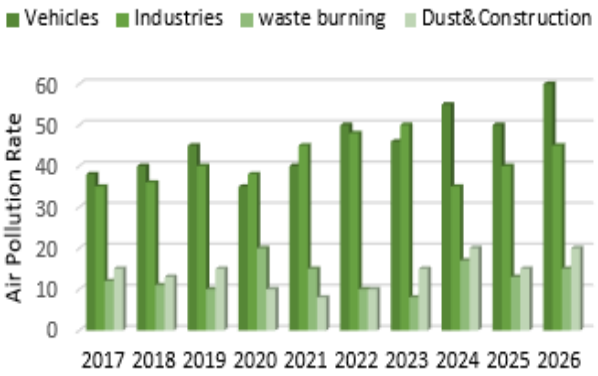


Figure 1. Air Pollution Rate

Figure 1 Shows the air pollution rate of past and future few years. According to the graph air pollution due to vehicles increases rapidly. The most important pollutants are vehicles, industries, waste burning, and dust & construction. Year by year usage of vehicles increases and air pollution also increases. Nowadays vehicles are one of the mandatory things in our daily day to life [9]. Vehicle emissions cause the discharge of many pollutants into the atmosphere, such as particulate matter (PM), carbon monoxide (CO), nitrogen oxides (NOx), and volatile organic compounds (VOCs). These contaminants can cause respiratory issues, poorer air quality, and the production of smog [10].

In addition, the release of greenhouse gases resulting from industrial processes amplifies the effects of climate

change, causing severe weather phenomena and disruptions to the ecosystem. Climate change, which is connected to greenhouse gas emissions from air pollution, exacerbates the issue. Consequently, there is an increased frequency of catastrophic weather occurrences and ecological imbalances [11,12]. To overcome these issues, a novel air quality monitoring using IoT has been proposed. The following is a list of the paper's main contributions.

- Initially, Sensors in the traffic system collect the environmental data and transmit it to the IoT gateway. It processes the data and gives it to the real-time data analytic module (RTAM).
- In the real-time data analytic module, the PCA technique is used to extract the features and it is transmitted to the DenseNet to predict the quality of the air. Again, predicted data is given to the RTAM.
- RTAM gives the predicted data to the data storage where all the data about the quality of the air is stored and to the center. If the value of polluted air is above the fixed threshold value it gives an alert and suggests the route to the user.
- We contrast the performance of the suggested model with other related strategies. The outcomes of our experiments were carried out in accordance with a thorough set of evaluation criteria,

The remainder of this research is explained as follows: Section II examines the study of the literature. Section III describes the proposed system in great depth. Section IV is the result and discussion, and Section V is the conclusion.

2. LITERATURE SURVEY

Several studies have utilized several techniques to monitor the quality of air in real time. The following section covers a few of the current evaluation approaches along with their disadvantages are as follows:

In 2022, Asha, P., et.al.,[14] suggested an Artificial Intelligence-based Environmental Toxicology for Air Pollution Monitoring System facilitated by the Internet of Things (ETAPM-AIT). The primary drawback is that ETAPM-AIT systems rely on dependable internet access for data transfer and communication. To assess the efficacy of the suggested ETAPM-AIT model, a comprehensive series of simulation analyses is conducted and the outcomes are reviewed after 5, 15, 30, and 60 minutes.

In 2022, Haq, M.A., [15] suggested the novel air pollution classification model SMOTEDNN (Synthetic Minority Oversampling Technique with Deep Neural Network). The primary performance issue arises from rigorous pre-processing of the data and comprehensive hyperparameter optimization. In terms of accuracy, the unique model SMOTEDNN performed better than the other models from the current study and previous research, with a score of 99.90%.

In 2022, Jabbar, W.A., et.al., [16] suggested the implementation of an outdoor-based LoRaWAN-IoT-AQMS (long-range wide area network-based Internet of Things air quality monitoring system). The primary drawback is the high network strain caused by the sensing node's constant data transmission to the cloud every 10 seconds. By contrasting the created LoRaWAN-IoT-AQMS results with experimental data from state-of-the-art Aeroqual air quality monitoring apparatus, the results are verified.

In 2022, Alvear-Puertas, V.E., et.al., [17] suggested the development of a portable, high-tech air-quality monitoring system that can assess local air pollution. Provide a suitable Internet of Things architecture with an Edge-based time series database, MQTT, and a lightweight messaging protocol. The IoT nodes utilized to infer air quality had a performance rate of more than 90% in terms of pertinent data. Moreover, the memory consumption 14 Kbytes in flash and 3 Kbytes in RAM was lower in terms of bandwidth and power requirements.

In 2022, Zhu, Y., et.al., [18] suggested an improved, inexpensive, Internet of Things-based IAQ monitoring system that uses artificial intelligence to generate recommendations. The LSTM AI technique is used to forecast future CO2 levels based on the collected CO2 data. This activity is limited by the fact that accurate and dependable measurements are dependent on routine sensor calibration and maintenance. The proposed approach can forecast the steady state of CO2 with a margin of error of 5.5%.

In 2023, Guerrero-Ulloa, G., et.al., [19] suggested Ide Air, an inexpensive Internet of Things-based system for monitoring air quality. Ide Air was designed to detect the levels of hazardous gases in indoor environments and, in response, to trigger alerts and messages, unlock doors, or activate fans. Ide Air was developed using the TDDM4IoTS technique, which aided the developers in completing IoTS development chores more quickly. Early results show that Ide Air is running with a high degree of acceptance.

In 2023, Pant, J., et.al., [13] suggested an intelligent fuzzy-based indoor air quality monitoring system based on the Internet of Things (IoT). The proposed system uses sensors to gather data in real-time on-air quality measurements, including a PM10 and CO2 sensor. The suggested system's main drawback is that, when utilizing several sensors, it could use a lot of energy. The results of the experiment demonstrate how effective the recommended strategy is for tracking and enhancing indoor air quality.

However, several related studies have been conducted to monitor the quality of the air. Moreover, there is a number of disadvantages in the existing methods like usage of more energy, high network load, computational complexity etc. This paper proposed a technique to eliminate these disadvantages, which is explained in the following session.

3. PORPOSED METHOD

The In this session, a novel air quality monitoring using IoT has been proposed to monitor the quality of the air in real-time. Initially, Sensors in the traffic system collect the environmental data and transmit it to the IoT gateway. It processes the data and gives it to the real-time data analytic module (RTAM). In RTAM feature extraction process is

done by using the PCA technique. The feature-extracted data is transmitted to the DenseNet to predict the quality of the air. Again, predicted data is given to the RTAM which is transferred to the data storage where all the data about the quality of the air is stored and to the control center. In the control center, it checks the value of pure and impure air. If the value of impure is above the fixed threshold value it gives an alert and suggests the route to the user and passes the information to the pollution control authority. The overall workflow of the proposed method is given in Figure 2.

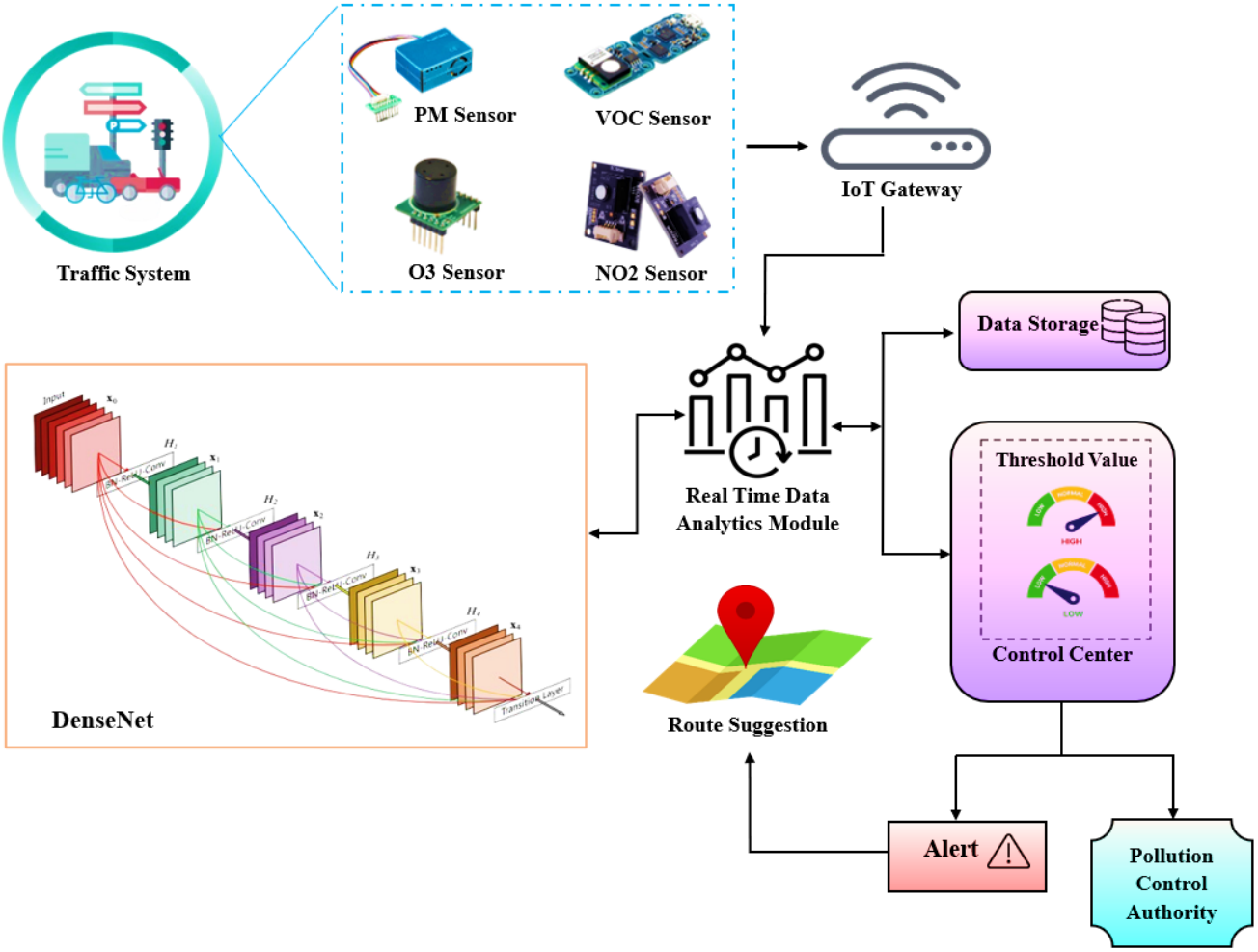


Figure 2. The Overall workflow of Proposed Methodology

3.1. Data Collection

Traffic System employs various sensors, including PM (particulate matter), VOC (volatile organic compounds), O3 (ozone), and NO2 (nitrogen dioxide) sensors collecting pollution data from a environment of urban areas. PM Sensor used to measure particulate matter (PM) concentration in the air, VOC Sensor used to detect VOCs present in the environment, O3 Sensor used to monitor ozone (O3) levels and NO2 Sensor used to measure nitrogen dioxide (NO2) concentration. These sensors are connected to an IoT Gateway, which transmits the collected data to a Real-Time Data Analytics Module.

3.2. Real-Time Data Analytics Module

The real-time data analytics module works in tandem to gather information on different pollutants, including nitrogen dioxide (NO2), CO, PM, ozone (O3), and sulphur dioxide (SO2). This module does the feature extraction of the sensor data. Sensor data frequently comes in raw and noisy formats, making it unsuitable for direct analysis or for inclusion in machine learning algorithms. In feature extraction, key patterns and data from the measurements are extracted from the raw sensor data and converted into a more concise and comprehensible representation. Features of Datus collected from the sensors are extracted in the real-time data analytic module. The Principal Component Analysis (PCA) is used for extracting features from the sensor data.

3.2.1. Principal Component Analysis (PCA)

PCA is a common method for reducing feature dimensionality. However, it is limited in complex feature spaces due to its linear nature. To address this, standard PCA is extended to nonlinear dimension reduction. Once features are normalized, PCA starts to be a helpful method. To minimize dimensionality in huge datasets, it finds the covariance matrix's eigenvectors with the largest eigenvalues. The definition of PCA algebraic is as follows: Calculate the mean of C for data outline C as follows:

$$\theta = M(C) \quad (1)$$

Determine C's covariance as follows:

$$CU = C_{ov}(C) = M[(C-\theta)(C-\theta)^T \quad (2)$$

Count the eigenvalue θ_i, and eigenvector b_1, b_2,……...b_N, i= 1, 2……F of the covariance CoV. For the Covariance, the equation is solved CoV;

$$V_k = \frac{\sum_{j=1}^{L}\theta_f}{\sum_{j=1}^{N}\theta_f} \quad (3)$$

Information regarding a more compact measurement subspace can be found by selecting the first L eigenvalue that achieved the desired mutual range, which should be 83% larger than the size of the major segments.

$$g = X^t - V \quad (4)$$

Where V is the first data set to be knotted, and t represents the transfer matrix. Operating the main L eigenvector independently from n to K ($K \ll n$).) increases the number of variables or measurements.

$$|\theta l - C0V| = 0 \quad (5)$$

However, l For having dimensions that are more than CoV, give the identity matrix the benefit of the doubt. Determine the θ_f Eigenvalues of component L by calculating the percentage of data that is accounted for by the first component.

3.3. Dense Net

DenseNet (Densely Connected Convolutional Networks) is a deep learning architecture that introduces dense connectivity between layers. It is particularly effective in addressing the vanishing gradient problem and encourages feature reuse throughout the network. These dense connections enable information flow directly from early layers to later layers, facilitating better gradient flow, deeper networks, and improved feature propagation. The key idea behind DenseNet is a dense connection pattern in which every layer in a dense block receives feature maps from all layers that come before it and transmits its own feature maps to all layers that come after it. Because every layer has access to the feature maps created by every layer before it, this leads to feature reuse and helps the network learn more discriminative features.

A DenseNet consists of several dense blocks that make up the network. Multiple convolutional layers with batch normalization and a non-linear activation function (usually ReLU) make up each dense block. The output of each layer within the dense block is concatenated with the feature maps of all preceding layers and fed as input to the subsequent layers within the same dense block. The DenseNet Block can be represented as given in equation (6)

$$y_{l+1} = E_l([y_0, y_1, \dots., y_l]) \quad (6)$$

Here, $[y_0, y_1, \dots., y_l]$ denotes the concatenation of the feature map from all the preceding layers. The transformation $E_l(\cdot)$ typically consists of a series of operations such as batch normalization, then a non-linear activation function (ReLU, for example), and then a convolution operation which is given in equation (7)

$$E_l(y) = RELU\big(BN(M_l \times y)\big) \quad (7)$$

Here, M_l represents the weights of the convolution operation, × stands for batch normalization, represents the convolution procedure, and stands for the rectified linear unit activation function (ReLU). Transition layers are included in between dense blocks to limit the expansion of feature maps and lower computational complexity. These layers include a batch normalization step, a 1x1 convolution operation for dimensionality reduction, followed by average pooling which is given in equation (8)

$$y' = AvgPool(E([y_0, y_1, \dots., y_{l-1}]) \quad (8)$$

Here, y' represents the output feature maps after passing through the transition layer, and AvgPool denotes the average pooling operation. Overall, DenseNet facilitates feature reuse and enables the network to be more parameter-efficient compared to traditional architectures, leading to improved performance, especially in tasks with limited training data. The output of the DenseNet is classified into two classes as pure and impure. These data are again sent back to the real time data analytic module where these data are transferred to the data storage and control centre. The data stored in the data storage for the purpose of future use.

3.4. Control Centre

In control center it checks the value of pure and impure air. When a process deviates from its expected operating range, these thresholds are predetermined boundaries or levels that are used to initiate particular actions or alerts. A control center's principal objective is to make sure that processes remain within reasonable bounds, maintain efficiency, and guard against problems or breakdowns. If the value of impure is above to the fixed threshold value it gives alert and suggest the route to the user and pass the information to pollution control authority.

4. RESULTS AND DISCUSSIONS

The proposed method's experimental results are analyzed and a discussion of performance is done in terms of numerous evaluation metrics within this section. The proposed framework is developed and assessed using the Python programming language along with libraries (such as sci-kit-learn, TensorFlow, Kera's, NumPy, and HDF5) on a Windows operating system with an Intel Core i7 CPU and 16GB RAM. The effectiveness of the suggested method is evaluated in this paper using the City Pulse EU FP7 Project's pollution dataset. The proposed model's effectiveness is contrasted with ETAPM-AIT [14], SMOTEDNN [15], and Ide Air [19]. Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), Accuracy, Precision, F1-Score, and time efficiency are used to assess the performance of the suggested Air IoT approach.

4.1. Description of dataset

The City Pulse EU FP7 Project's pollution dataset, has 8 features total, was used in the experiment. These features are ozone, carbon monoxide, particulate matter, Sulphur dioxide, longitude, latitude, nitrogen dioxide, and timestamp. The 17568 samples in the dataset were taken at intervals of five

minutes. EPA's AQI standard is presented for each sample value.

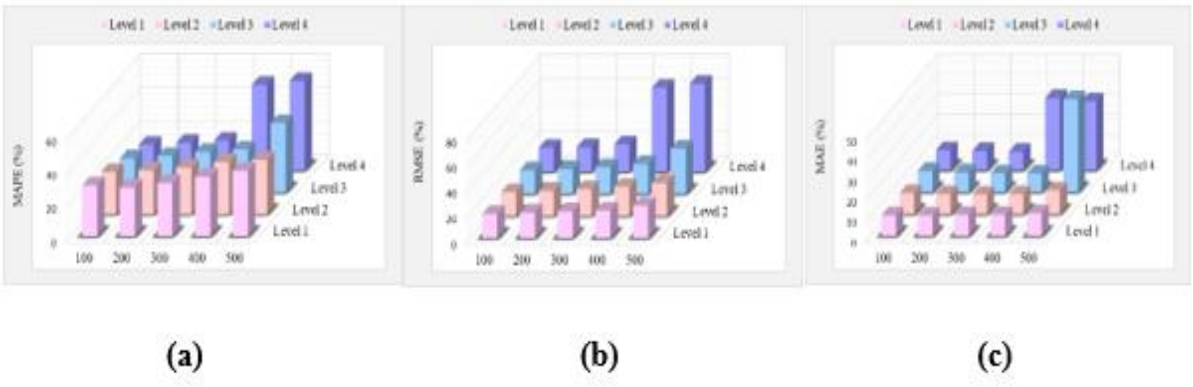


Figure 3. Performance Across Different Network Levels (a) MAPE (b) RMSE (c) MAE

The performance of various levels of MAPE, RMSE, and MAE across the network is displayed in Figure 3. It shows that performance can be somewhat improved by adding extra nodes after each layer has 300 nodes. Our model demonstrated the best performance. Adding more nodes to each layer would lead to overfitting and an unnecessarily long training period. Using a four-layer layout with 400 or 500 nodes in each layer, these can be easily illustrated because the validation error builds up quickly.

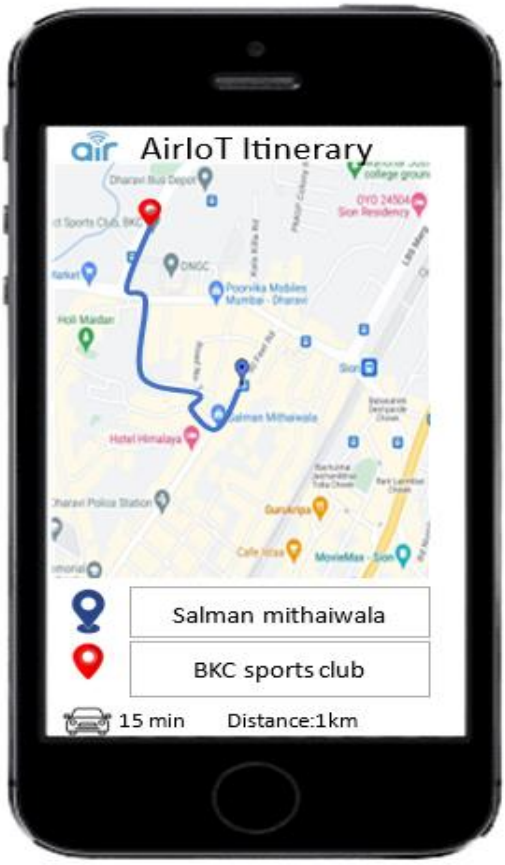


Figure 4. Android Application showing Pollution less Route

The recommended route in a low-pollution area is shown in Figure 4. The amount of pollution throughout the entire path for a user traveling from a source to a destination will be forecasted, and if the quantity is excessive, a warning will be displayed so the user can reroute his travel. The proposed map provides the user with an alternate path to the location where air pollution is at a minimum.

In Figure 5, the proposed technique, and the existing method such as ETAPM-AIT [14], SMOTEDNN [15], and Ide Air [19] are contrasted for accuracy using City Pulse EU FP7 Project's pollution dataset. Accuracy is a crucial element that illuminates the evaluation of a particular classifier's performance. The accuracy of the Air IoT technique is increased by 10.08%, 17.64% and 34.34% as compared to the ETAPM-AIT, SMOTEDNN, and Ide Air methods.

Figure 5. Performance Comparison in terms of accuracy

Figure 6. Performance of models on dataset

Figure 6 shows the performance comparison of the proposed Air IoT method and the existing ETAPM-AIT [14], SMOTEDNN [15], and Ide Air [19] methods in terms of accuracy, precision and F1-score using City Pulse EU FP7 Project's pollution datasetThe accuracy of the proposed system is increased by 10.08%, 17.64% ,34.34% and the precision is increased by 9.59%, 18.56%, 17.93% and the F1-score is increased by 8.79%, 16.96%, 36.85% as compared to the Ide Air, SMOTEDNN, and ETAPM-AIT methods respectively.

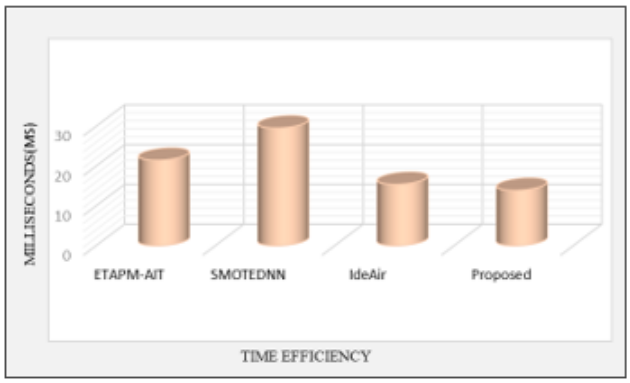


Figure 7. Performance Comparison in terms of time efficiency

Figure 7 displays the time efficiency of the proposed Air IoT technique and existing ETAPM-AIT [14], SMOTEDNN [15], and Ide Air [19] methods. How quickly and effectively the system can gather, process, and disseminate information concerning air pollution levels is referred to time efficiency. The proposed system's time of 13.87 milliseconds is relatively quick compared to ETAPM-AIT, SMOTEDNN, and Ide Air techniques which take 21.45 milliseconds, 29.56 milliseconds, and 15.39 milliseconds, respectively. It shows that the proposed technique takes less time to process compared to the existing methods.

5. CONCLUSION

In this paper, a novel air quality monitoring using IoT has been proposed to monitor the quality of the air in real time. The quality of air is predicted using DenseNet Model which classifies the quality into 3 classes as pure, impure, and normal. The proposed system's effectiveness is assessed using The City Pulse EU FP7 Project's pollution dataset. The proposed framework is developed and assessed using the Python programming language The performance of the proposed method is evaluated by MAE, RMSE, MAPE, Accuracy, Precision, F1-Score and time efficiency. According to the comparative analysis, the accuracy of the proposed system is increased by 10.08%, 17.64% and 34.34% as compared to the Ide Air, SMOTEDNN, and ETAPM-AIT methods respectively. Future research may concentrate on offering hyper-localized air quality predictions rather than merely city-wide or regional forecasts. This can entail creating models that consider localised human activity and microclimate conditions.

CONFLICTS OF INTEREST

The authors declare that there is no conflict of interest.

FUNDING STATEMENT

Not Applicable.

ACKNOWLEDGEMENTS

The author would like to express his heartfelt gratitude to the supervisor for his guidance and unwavering support during this research for his guidance and support.

REFERENCES

[1] G. Marques, N. Miranda, A. Kumar Bhoi, B. Garcia-Zapirain, S. Hamrioui and I. de la Torre Díez, “Internet of things and enhanced living environments: measuring and mapping air quality using cyber-physical systems and mobile computing technologies”, *Sensors*, vol. 20, no. 3, pp. 720, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[2] T. Alam, “Cloud-based IoT applications and their roles in smart cities,” *Smart Cities*, vol. 4, no. 3, pp. 1196-1219, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[3] S.A. Meo, F.J. Almutairi, A.A. Abukhalaf, O.M. Alessa, T. Al-Khlaiwi and A.S. Meo, “Sandstorm and its effect on particulate matter PM 2.5, carbon monoxide, nitrogen dioxide, ozone pollutants and SARS-CoV-2 cases and deaths”, *Science of the Total Environment*, vol. 795, pp. 148764, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[4] V. Barot, V. Kapadia and S. Pandya, “QoS enabled IoT based low-cost air quality monitoring system with power consumption optimization”, *Cybernetics and Information Technologies*, vol. 20, no. 2, pp. 122-140, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[5] M. Sahoo and N. Sethi, “The dynamic impact of urbanization, structural transformation, and technological innovation on ecological footprint and PM2. 5: evidence from newly industrialized countries”, *Environment, Development and Sustainability*, vol. 24, no. 3, pp. 4244-4277, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[6] I. Manisalidis, E. Stavropoulou, A. Stavropoulos, and E. Bezirtzoglou, “Environmental and health impacts of air pollution: a review”, *Frontiers in public health*, vol. 8, pp. 14, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[7] E. Hernandez-Rodriguez, D. Kairúz-Cabrera, A. Martinez, R.A. González-Rivero and O. Schalm, “Low-Cost Portable System for the Estimation of Air Quality”, *In The conference on Latin America Control Congress*, pp. 287-297, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[8] J. Chatkin, L. Correa, and U. Santos, “External environmental pollution as a risk factor for asthma”, *Clinical reviews in allergy & immunology*, pp. 1-18, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[9] A. Gil, “Challenges on waste-to-energy for the valorization of industrial wastes: Electricity, heat and cold, bioliquids and biofuels”, *Environmental Nanotechnology, Monitoring & Management*, vol. 17, pp. 100615, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[10] A. Dandotiya, and H.K. Sharma, “Climate change and its impact on terrestrial ecosystems”, *In Research Anthology on Environmental and Societal Impacts of Climate Change*, pp. 88-101, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[11] W. Al-Delaimy, V. Ramanathan, and M. Sánchez Sorondo, “Health of people, health of planet and our responsibility: Climate change, air pollution and health”, pp. 419, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[12] K.L. Ebi, J. Vanos, J.W. Baldwin, J.E. Bell, D.M. Hondula, N.A. Errett, K. Hayes, C.E. Reid, S. Saha, J. Spector, and P. Berry, “Extreme weather and climate change: population health and health system implications”, *Annual review of*

public health, vol. 42, no. 1, pp. 293-315, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[13] P. Asha, L.B.T.J.R.R.G.S. Natrayan, B.T. Geetha, J.R. Beulah, R. Sumathy, G. Varalakshmi and S. Neelakandan, "IoT enabled environmental toxicology for air pollution monitoring using AI techniques," *Environmental research*, vol. 205, pp. 112574, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[14] M.A. Haq, "SMOTEDNN: A novel model for air pollution forecasting and AQI classification", *Computers, Materials & Continua*, vol. 71, no. 1, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[15] W.A. Jabbar, T. Subramaniam, A.E. Ong, M.I. Shu'Ib, W. Wu and M.A. de Oliveira, "LoRaWAN-based IoT system implementation for long-range outdoor air quality monitoring", *Internet of Things*, vol. 19, pp. 100540, 2022[CrossRef] [Google Scholar] [Publisher Link]

[16] V.E. Alvear-Puertas, Y.A. Burbano-Prado, P.D. Rosero-Montalvo, P. Tözün, F. Marcillo and W. Hernandez, "Smart and Portable Air-Quality Monitoring IoT Low-Cost Devices in Ibarra City, Ecuador", *Sensors*, vol. 22, no. 18, pp. 7015, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[17] Y. Zhu, S.A. Al-Ahmed, M.Z. Shakir and J.I. Olszewska, "LSTM-based IoT-enabled CO2 steady-state forecasting for indoor air quality monitoring", *Electronics*, vol. 12, no. 1, pp. 107, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[18] G. Guerrero-Ulloa, A. Andrango-Catota, M, Abad-Alay, M.J. Hornos, and C. Rodríguez-Domínguez, "Development and assessment of an indoor air quality control IoT-based system", *Electronics*, vol. 12, no. 3, pp. 608, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[19] J. Pant, H. Pant, D. Rautela, P. Sethuramaligam, R. Iyer and V. Birchha, "automated indoor air quality monitoring using an intelligent IoT fuzzy-based approach". [CrossRef] [Google Scholar] [Publisher Link]

AUTHORS

M. Devaki completed UG in Electrical & Electronics Engineering from Raja College of Engineering & Technology, Anna University, Tirunelveli during the year 2011with 82% (First Class with Distinction), completed PG in Power Systems from Velammal College of Engineering & Technology, Anna University, Chennai during the year 2015 with 8.59 CGPA (First Class with Distinction). Currently pursuing Ph.D. in Electrical Engineering, Anna University, Chennai. She has 12 years of teaching experience. Currently working as Assistant Professor in the Department of EEE, Velammal College of Engineering & Technology, Madurai, Tamilnadu, India. Her research area includes Power Quality Improvement, Monitoring, Renewable Energy Sources and Internet of Things (IoT).

Jeyaraman Sathiamoorthy is currently working as a Professor RMK Engineering College, Kavarapettai in Chennai. He has completed M. Tech (CIT) and Ph. D from Manonmaniam Sundaranar University, Tirunelveli. He has 20 years of teaching experience and she has published papers in Network Security in National and International journals and has presented in International Conferences and Seminars. His current area of interest is Programming Languages, Algorithms and ad-hoc networks especially MANET, VANET, FANET and Underwater Communication.

M. Usha is currently working as a Professor cum Assistant Director in MEASI Institute of Information Technology, Chennai. She has completed M.C.A. and M. Phil in Computer Science from Bharathidasan University, Trichy. She has also done her M. Tech (CIT) and Ph. D from Manonmaniam Sundaranar University, Tirunelveli. She has 20 years of teaching experience and she has published papers in Network Security in National and International journals and has presented in International Conferences and Seminars. Her current area of interest is Operating Systems, Algorithms and ad-hoc networks especially MANET, VANET, FANET and Underwater Communication.

Arrived: 23.07.2024
Accepted: 22.08.2024

International Journal of Data Science and Artificial Intelligence (IJDSAI)
Volume 02, Issue 04, July – August (2024)

RESEARCH ARTICLE

CLASSIFICATION OF LIVER CANCER VIA DEEP LEARNING BASED DILATED ATTENTION CONVOLUTIONAL NEURAL NETWORK

R. Ramani [1, *], K. Vimala Devi [2], P. Thiruselvan [3] and M. Umamaheswari [4]

[1] Department of Computer Science and Engineering, P.S.R Engineering College, Sivakasi, Tamil Nadu 626140, India.
[2] School of computer Science and Engineering, Vellore Institute of technology, Vellore, Tamil Nadu, 632014 India.
[3] Department of Computer Science and Engineering, P.S.R Engineering College, Sivakasi, Tamil Nadu 626140, India.
[4] Department of Computer Science and Engineering, P.S.R Engineering College, Sivakasi, Tamil Nadu 626140, India.

[*]*Corresponding e-mail: rramani.ananth@gmail.com*

Abstract – Liver cancer occur when normal cells develop aberrant DNA alterations and reproduce uncontrollably. Patients with cirrhosis, hepatitis B or C, or both have an increased risk of developing the progressing stage of cancer. The radiologists spend more time for detecting the liver cancer when analysing with traditional methods. Early detection of liver cancer can help doctors and radiation therapists identify the tumours. However, manual identification of liver cancer is time-intensive and challenging process in the current scenario. In this work, an automated deep learning network is designed to classify the liver cancer in its initial phase. At first, the CT scans are gathered from the publicly available LiTS database and these gathered images are pre-processed using Gaussian filter is used for reducing the noises and to smoothen the edges. The liver region is segmented using Enhanced otsu (EM) method is utilized to segment the liver region separately from the pre-processed input images. Afterwards, Dilated Convolutional Neural Network (DCNN) with the attention block is employed for classifying the liver cancer into tri-classes such as normal controls (NC), hepatocellular carcinoma (HCC) and cholangiocarcinoma (CC) cases based on the extracted features. The efficiency of the proposed DA-CNN is evaluated using the attributes viz., accuracy, sensitivity, precision, specificity, and F1-score values are computed as classification results. The experimental fallouts disclose that the DA-CNN attains an accuracy range of 98.20%. Moreover, the proposed DA-CNN advances the overall accuracy by 3.25%, 5.29%, and 0.99% better than Optimised GAN, OPBS-SSHC, HFCNN respectively.

Keywords *– Liver cancer, Deep learning, CT images, Attention block, Enhanced otsu method.*

1. INTRODUCTION

Liver cancer (LC) is a malicious tumour that originates in the liver cells, medically referred to as hepatocellular carcinoma (HCC). Globally, LC is the prevalent and aggressive forms of cancer with a high mortality rate [1]. LC can be caused by chronic liver diseases like hepatitis B or C infection, cirrhosis, heavy liquor consumption, exposure to aflatoxins, and certain genetic conditions [2, 3]. In most cases, LC progresses silently, with symptoms frequently appearing only after the condition has progressed. In addition to fatigue, jaundice, and unexplained weight loss, stomach pain is another symptom. LC poses significant challenges due to its insidious onset, complex etiology, and limited treatment options, resulting in a high mortality rate [4, 5].

Regular screenings and timely medical intervention are essential for early detection, which improves prognosis and increases the efficacy of existing treatment modalities. Diagnostic procedures often include imaging studies, blood testing, and occasionally a liver biopsy [6, 7]. Depending on the cancer's stage, a patient may receive chemotherapy, radiation therapy, targeted therapy, surgery, or a liver transplant among other treatments [8]. The prognosis for liver cancer is still poor despite improvements in treatment options, underscoring the vital significance of early intervention, routine screening, and preventative measures. The precision and efficiency of traditional diagnostic procedures such as MRI, CT and US imaging techniques are limited which frequently results in incorrect or delayed diagnosis [9, 10].

Machine learning (ML) [11] and Deep learning (DL) [12] has emerged as a promising tool in medical imaging analysis, including the identification and diagnosis of LC. DL algorithms [13] can effectively learn intricate patterns and features from vast amounts of medical imaging data, enabling them to detect subtle abnormalities indicative of liver cancer with high accuracy and efficiency [14]. In this research, we explore the application of DL technique for the detection of liver cancer using CT scans. The primary contributions of the work are summarised as:

- Initially, the CT images are gathered and pre-processed using Gaussian filter is used for reducing the noises and to smoothen the edges.
- An Enhanced otsu (EM) method is used to segment the liver region separately from the pre-processed CT images.
- Afterwards, Dilated Convolutional Neural Network (DCNN) induced with attention block is employed for classifying the liver cancer into normal controls (NC), hepatocellular carcinoma (HCC) and cholangiocarcinoma (CC) cases.
- The efficiency of the proposed DA-CNN is evaluated using the attributes viz., accuracy, sensitivity, precision, specificity, and F1 score for computing the classification results.

The rest of the paper was scheduled as follows in advance. A summary of the literature was provided in Section 2, followed by an extensive description of the proposed DA-CNN methodology for LC classification in Section 3, results and discussion in Section 4, and Section 5 holds the conclusion part.

2. LITERATURE REVIEW

In this section, the challenges associated with traditional diagnostic methods was discussed and spotted the use of DL-based approaches in overcoming these challenges. Furthermore, the recent advancements and existing architectures in DL was reviewed for LC recognition, emphasizing their strengths and limitations in this section.

In 2022, Amin et al., devised [15] an optimized GAN for image synthesis, followed by localization using an improved model. Deep features from pre-tuned ResNet50 were inputted to the YOLO-v3 model. Segmentation employs a pre-trained InceptionResNetV2 model for Deeplabv3, fine-tuned with annotated masks. Experimental findings demonstrate a testing accuracy exceeding 95%.

In 2021 Kushnure, and Talbar putforward [16] a multi-scale approach that augmented the CNN's receptive field by incorporating multi-scale features, thereby capturing both local and global characteristics at a better granularity. Experimental results demonstrated improved efficiency of the system on the 3Dircadb database. Specifically, the method attained a dice score of 97.1% and 84.1% for LC detection.

In 2021 V. Hemalatha et al., [17] introduced a method that combines Region of Interest (ROI) extraction with the Adaptive WS technique for the detection of LC. This methodology incorporates ANN techniques for denoising, scanning, extraction, and segmentation. To recognize LC within real-time datasets, a feed-forward neural network was employed. Subsequently, the extraction of features was conducted using GLCM techniques.

In 2020 B. Sakthisaravanan et al. [18] designed an OPBS-SSHC approach for liver tumor identification, integrating segmentation and similarity-based hybrid classification. Noise removal during preprocessing was followed by edge enhancement using a frequency-centered sharpening technique. Subsequently, the SSHC model classified extracted features, achieving a superior accuracy of 93% compared to other systems.

In 2020 Dong, et al., designed [19] developed a Hybridized Fully CNN for the segmentation of LC. The suggested method combines residual and pre-trained weights with the effectively extracted features from Inception. This DL system illustrates the idea of illuminating certain decision-making steps in a deep neural network was trained extensively. From the analysis the suggested HFCNN attains the accuracy of 97.22% for 50 epochs.

In 2019 Hamm., et al., [20] devised custom convolutional neural network (CNN) through iterative refinement of the network architecture and training samples. The final CNN comprised three convolution layers with Relu, two max pooling layers, and two fully connected layers. Monte-Carlo cross-validation was employed during this method development process. Upon completion of model engineering, the classification accuracy of the finalized CNN reached 92.0%.

According to the literature review, the existence of noise abnormalities in CT scans poses challenges for liver segmentation. Given the intricate structure of the liver, many clinical decisions support schemes, particularly those employing ML techniques, rely heavily on segmentation. Utilizing CT scan images for automated LC diagnosis is pivotal due to potential variations in structural alterations among patients. To address these issues, a novel DA-CNN model utilizing CT images for early-stage LC identification has been proposed. This study introduces a region-based segmentation technique for recognizing the LC area, ultimately aiming to design an efficient approach for LC categorization.

3. PROPOSED METHOD

This proposed section presents a novel deep learning-DA-CNN model to identify the LC cases from the available LiTS dataset. The overall workflow of the proposed LC identification method is displayed in Figure 1.

3.1. Dataset description

This study utilizes the common LiTS dataset from [21] which comprising 194 CT scans containing lesions and 201 liver CT images. The diverse and varied characteristics of tumor lesions pose significant challenges for automated segmentation. The objective is to devise segmentation techniques capable of automatically detecting liver tumours in contrast-enhanced CT scans. The test dataset comprises 70 CT scans, while the training dataset consists of 130 CT scans. This task is coordinated with MICCAI 2017 and ISBI 2017.

3.2. Gaussian filter

Pre-processing of medical images is essential for improving the quality and interpretability of diagnostic results. The Gaussian filter is a linear filter which reduces noise and smooths images while maintaining important features. In the image, each pixel is subjected to a weighted average determined by a Gaussian function. A Gaussian

function smoothes the image by giving more weight to pixels near the center of the filter window.

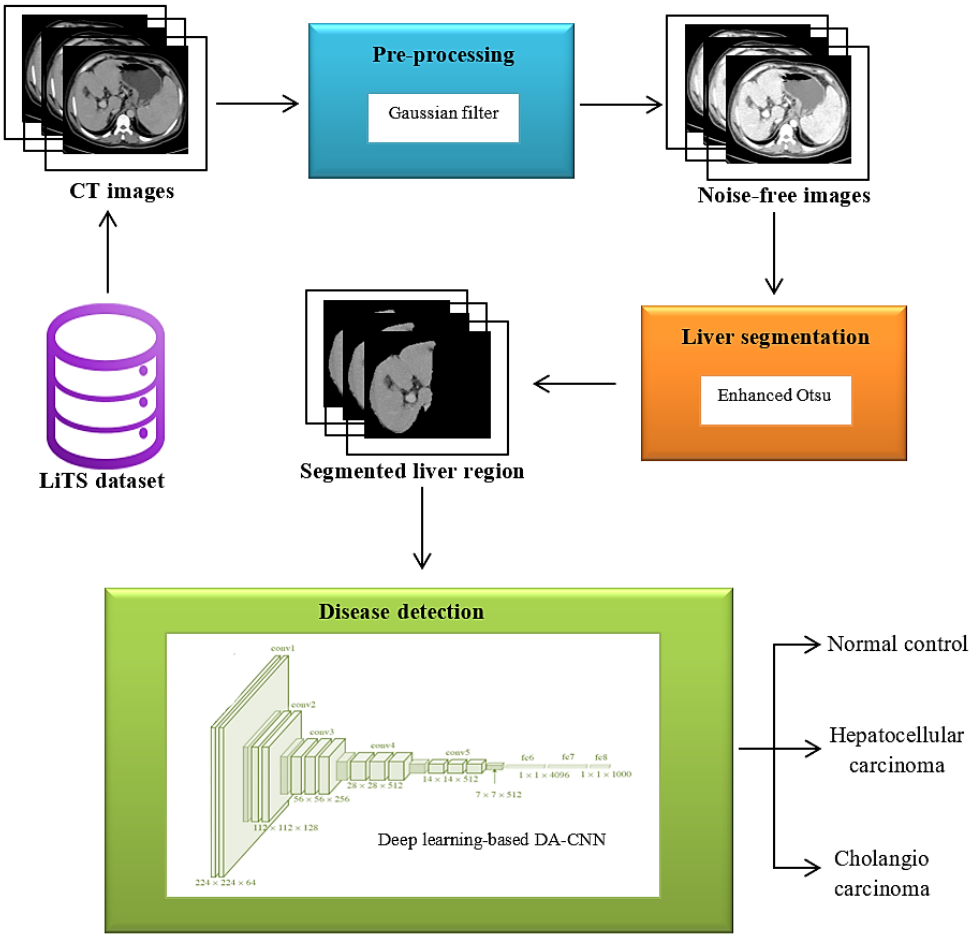


Figure 1. The outline of the overall proposed LC identification model

This gaussian function is mathematically determined as,

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (1)$$

Where $G(x,y)$ represents the Gaussian kernel, σ is the standard deviation of the Gaussian distribution, which controls the amount of smoothing applied to the image, *x* and *y* are the spatial co-ordinates of the filter. In the next step, the input is convolved by a Gaussian filter to create the smoothed output. Based on the specific requirements, the kernel size and σ value was changed in the properties of the input images.

3.3. Enhanced otsu method

After filtering the image is subjected to a segmentation process, where the Enhanced otsu (EM) method is used for liver region segmentation. In this EM method, the discriminate analysis is used where the threshold value is set in retaining the pixels of the image when it is split into two such classes. C_1 and C_2, where $C_1 = \{0,1,2,\dots t\}$ and $C2 = \{t+1, t+2, \dots M-1\}$.The values corresponding to the class variance (within and in between) and the total variance are analysed in such a manner that to determine very effectively. Here n_i is considered as the no.of pixels along with grey level (n) and (i), which is considered as the total no.of pixels in the image and is represented as

$$\Gamma = \sum_{\tau=0}^{M-1} \Gamma_i + \sum_{\sigma=1}^{N} T(n-1) \quad (2)$$

As per the condition, Pi is represented as the possibility of incidence of the modified grey level i. Otsu method used here will analyse the becoming aspects of the threshold values in order to analyse the obtained optimal values for the given image. The threshold value t of the given image will indicate the $C_1 = \{0,1,2,\dots t\} and C2 = \{t+1, t+2, \dots M-1\}$. This is mathematically expressed as

$$\Gamma_{rs}(x) = \sum_{x=1}^{\infty} x(k). \frac{1}{\vartheta\sqrt{\sigma_x}}. \; exp\left[\frac{P_k}{2\sigma_x}\right] \quad (3)$$

The maximum value of the segmented image is used for evaluating the classes C1 and C2 in a separate process. This could be possible by fixing the original image to a certain extent in combination with the histogram equalization methods. The Corresponding equation representing the output value of segmentation is given as

$$X_G[P]_\alpha = \sum_{x,y\in\gamma} \Gamma_{rs}(x-y) + \frac{1}{R_p} \sum_{x.y} [P_x - P_y] \quad (4)$$

Here the lower bound is gathered by the possibility of considering the original image with a single grey Constraints

from the lower bound and the upper bound in correlation with the images which is of different values.

3.4. Dilated Attention-Convolutional Neural Network

In DA-CNN, an attention block is integrated with DCNN for focusing the most relevant features while extracting for better classification results. DCNN introduce dilated convolutions, also known as atrous convolutions, which enable capturing larger receptive fields without increasing the number of parameters. Dilated convolutions enable a larger responsive region without adding more parameters. In liver CT images, DCNN excel at extracting these features by effectively capturing spatial relationships at different scales. By employing dilated convolutions with increasing dilation rates across multiple layers, the network can aggregate information from a wide range of spatial contexts, enabling it to discern subtle patterns indicative of liver lesions. In particular, the convolution process can be expressed as follows assuming input features X and a filter K as follows

$$(X_P * F_K)(l,k) = \sum_n \sum_m X_P\,(l - n * d_r\,, k - m * d_r) F_K(n,m) \quad (5)$$

Where, d_r be the dilated rate and F_{K1} filter captures larger patterns because it encompasses a broader range of features than the F_{K2} filter, which operates as a standard convolutional filter and is ideal for extracting few patterns.

$$U = n + (n-1) \times (D_i - 1) \quad (6)$$

The image integrating matrix U is continually scanned by the convolution kernel. Dilated convolutions increase the interval of scanning features and add a few areas among convolution kernels. The height of the equivalent convolution kernel, essentially determines the number of images expressed as U in equations (6), assuming that the dilation rate is Di.

$$c_n = X(W' \times s^k_{i:i+u-1} + d) \quad (7)$$

The feature c_n is retrieved and expressed as follows after the convolution procedure.

$$Q_i = K_m = V_m = o_p \quad (8)$$

The outcomes of the dilated convolution o_p are the beginning values for the query matrix Qi, key matrix K_m, and value matrix V_m, as illustrated in the above equation. The attention block includes improved focus on relevant spatial information, enhanced feature learning, and the ability to adaptively attend to different parts of the input data for better task performance. To learn a spatial weight map W_m and then multiply it by the associated spatial locations.

$$W_m(C) = sigmoid\big(c^k([Avgpool(C); Maxpool\,(C)])\big) \quad (9)$$

where c^k represents a convolution operation with kernel size k. This module produces a spatial attention-map by considering the importance of each spatial location within the feature maps. The spatial-attention mechanism is illustrated as,

$$M_s(f) = \sigma(f\left(f_j(f_{avg} + f_{max})\right)) \quad (10)$$

where f_j is the join operation, f_{avg} indicates global average pooling and f_{max} global max pooling features respectively. In order to calculate attention weights, self-attention mechanisms use relationships between multiple variables in the same input sequence. As a result, DCNN is particularly helpful for tasks following feature extraction, as it can efficiently collect long-range dependencies and contextual information. To minimize spatial dimensions and reduce the sample size of feature maps, the DA-CNN can additionally include pooling layers. After feature extraction, the dilated CNN can be utilized for classification tasks such as distinguishing between benign and malignant lesions or identifying specific types of liver abnormalities. The extracted features serve as rich representations of the input images, which was the input to fully connected layers for tri-LC classification.

4. RESULTS AND DISCUSSION

This section uses Matlab-2020b to implemented the experimental fallouts and assess the efficacy of the proposed DA-CNN. The CT images as input are gathered from the accessible LiTS dataset. The comparison provides a detailed description and analysis of the total accuracy rate besides the efficiency of the proposed DA-CNN is also provided in this section.

4.1. *Efficacy scrutiny*

The effectiveness of the proposed DA-CNN was calculated using the network parameters viz., precision (P), recall (R), F1 score (F1), accuracy (A), and specificity (S).

$$A = \frac{(TP+FP)}{(TP+TN+FN+FP)} \quad (11)$$

$$P = \frac{TP}{TP+FP} \quad (12)$$

$$S = \frac{TN}{TN+FP} \quad (13)$$

$$R = \frac{TP}{TP+FN} \quad (14)$$

$$F1 = 2\left(\frac{P*R}{P+R}\right) \quad (15)$$

where TP and TN means true positives and negatives of the images, FP and FN specifies false positives and negatives of the images. For the experimental setup, the tri classes of LC are defined as “class-0” for NC, “class-1” for HCC, and “class-2” for CC respectively. The competence of the DA-CNN model for classifying several forms of LC is tabulated in Table.1 and it is visually signified in Figure 2.

Table 1. Efficacy evaluation of our DCNN model

Classes	A	S	P	R	F1
class-0	98.62	97.18	97.21	97.14	97.62
class-1	97.44	96.23	96.18	97.32	97.25
class-2	98.55	97.02	97.22	98.08	98.43

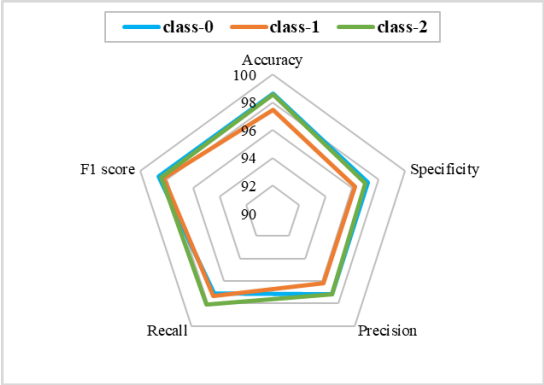


Figure 2. Classification performance analysis for tri-LC classes

The proposed DA-CNN model classified the three different LC classes from the CT images as shown in Figure 3 The proposed DA-CNN is assessed in term recall, accuracy, precision, specificity, and F1-score. The proposed DA-CNN achieves an overall accuracy of 98.20%. Also, the proposed DA-CNN exhibits an overall S of 96.81%, P of 96.87%, R of 97.51%, and an F1 of 97.76% respectively.

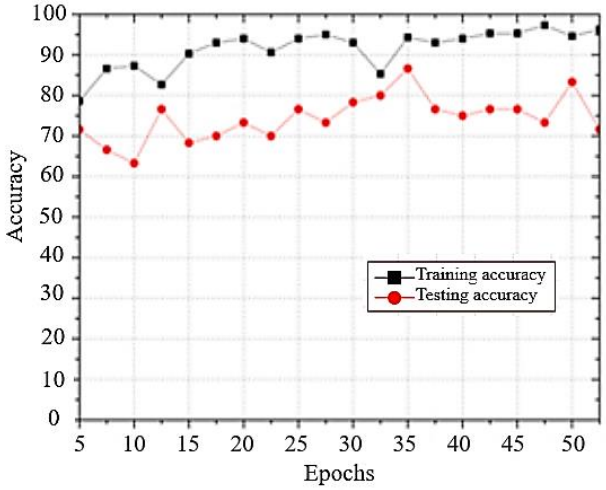


Figure 3. Accuracy graph of the proposed DA-CNN

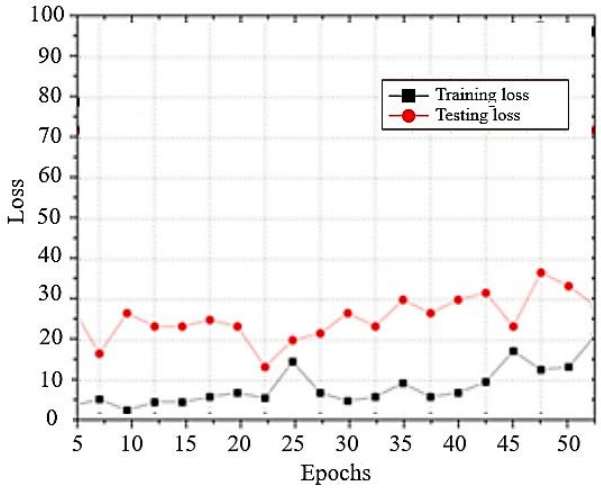


Figure 4. Loss graph of the proposed DA-CNN

Figure 3 depicts the accuracy curve, shows the accuracy range on the vertical axis against the number of epochs on the horizontal axis. When the no. of epochs raises, the proposed DA-CNN demonstrates an improvement in accuracy. Figure 4 shows the epochs and loss, illustrating that the DA-CNN experiences a reduction in loss with an increase in epochs. The proposed DA-CNN model proven effective in accurately classifying the tri-LC cases from the gathered CT images. According to the findings, the DA-CNN achieves significant performance in classification accuracy of 98.20%.

4.2. Comparative analysis

In this analysis, the competence of proposed and existing models was estimated using different metrics. The comparison assessment was competed among the proposed D-CNN with different classification techniques. The comparison of traditional classification networks is illustrated in Table.2.

Table 2. Comparison of traditional models for classification

Models	Precision	Recall	F1 score	Specificity	Accuracy
ANN [22]	87.3	84.5	87.6	85.4	84.5
SNN [23]	88.5	87.2	86.4	90.2	92.5
DACNN	96.8	97.5	97.7	96.8	98.4

Table 2 presents a comparison of various conventional DL networks, identifying the best classification accuracy achieved. Though, despite their utilization, classic DL networks didn't yield superior outcomes in comparison to the proposed DCNN. The proposed DCNN increases the overall accuracy by 11.0%, and 9.87% better than ANN and SNN respectively.

Table 3. Accuracy assessment among Proposed and Existing models

Authors	Methods	Accuracy
Amin., et al., [15]	Optimized GAN	95.0%
Sakthisaravanan, B. and Meenakshi [18]	OPBS-SSHC	93.0%
Dong., et al., [19]	HFCNN	97.22%.
Proposed model	DA-CNN	98. 42%

Table 3 illustrates the assessment of the DA-CNN model with prior models based on LiTS dataset. The proposed DA-CNN model advances the overall accuracy by 3.25%, 5.29%, and 0.99% better than Optimized GAN [15], OPBS-SSHC [18], HFCNN [19] respectively. Though, the existing networks not performed well when compared to the proposed network. So, the estimated fallouts of the proposed DA-CNN are extremely consistent for classifying the liver cancer in its early stages based on CT images from LiTS datasets.

5. CONCLUSION

This paper presents an automated deep learning DA-CNN for the identification of LC in its primary phases. The images are gathered from the publicly available LiTS database and these gathered images are pre-processed using Gaussian filter is used for reducing the noises and to smoothen the edges. The EM technique was used for segmenting the liver region separately from the pre-processed CT images. Afterwards, DA-CNN was employed for identifying the LC into tri-classes. The efficiency of the proposed DA-CNN is evaluated using the attributes like accuracy, sensitivity, precision, F1 score, and specificity values. The experimental fallouts disclose that the proposed DA-CNN attains an accuracy of 98.45%, that was comparatively better than the prior techniques. The proposed DCNN increases the overall accuracy by 11.0%, and 9.87% better than ANN and CNN respectively. Moreover, the proposed DA-CNN model advances the overall accuracy by 3.25%, 5.29%, and 0.99% better than Optimised GAN, OPBS-SSHC, HFCNN respectively. Therefore, the outcomes obtained from the proposed DA-CNN are highly trustworthy for the early-stage classification of liver cancer.

CONFLICTS OF INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

FUNDING STATEMENT

Not applicable.

ACKNOWLEDGEMENTS

The author would like to express his heartfelt gratitude to the supervisor for his guidance and unwavering support during this research for his guidance and support.

REFERENCES

[1] L. Huang, H. Sun, L. Sun, K. Shi, Y. Chen, X. Ren, Y. Ge, D. Jiang, X. Liu, W. Knoll, and Q. Zhang, "Rapid, label-free histopathological diagnosis of liver cancer based on Raman spectroscopy and deep learning", *Nature Communications*, vol. 14, no. 1, pp.48, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[2] A. Bakrania, N. Joshi, X. Zhao, G. Zheng, and M. Bhat, "Artificial intelligence in liver cancers: Decoding the impact of machine learning models in clinical diagnosis of primary liver cancers and liver cancer metastases", *Pharmacological Research*, vol. 189, pp.106706, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] S.P. Deshmukh, D. Choudhari, S. Amalraj, and P.N. Matte, "Hybrid deep learning method for detection of liver cancer", *Computer Assisted Methods in Engineering and Science*, vol. 30, no. 2, pp.151-165, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[4] J. Kim, J.H. Min, S.K. Kim, S.Y. Shin, and M.W. Lee, "Detection of hepatocellular carcinoma in contrast-enhanced magnetic resonance imaging using deep learning classifier: a multi-center retrospective study", *Scientific reports*, vol. 10, no. 1, pp.9458, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[5] C. Sun, A. Xu, D. Liu, Z. Xiong, F. Zhao, and W. Ding, "Deep learning-based classification of liver cancer histopathology images using only global labels", *IEEE journal of biomedical and health informatics*, vol. 24, no. 6, pp.1643-1651, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[6] B.C. Anil, P. Dayananda, B. Nethravathi, and M.S. Raisinghani, "Efficient Local Cloud-Based Solution for Liver Cancer Detection Using Deep Learning", *International Journal of Cloud Applications and Computing (IJCAC)*, vol. 12, no. 1, pp.1-13, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[7] S.H. Zhen, M. Cheng, Y.B. Tao, Y.F. Wang, S. Juengpanich, Z.Y. Jiang, Y.K. Jiang, Y.Y. Yan, W. Lu, J.M. Lue, and J.H. Qian, "Deep learning for accurate diagnosis of liver tumor based on magnetic resonance imaging and clinical data", *Frontiers in oncology*, vol. 10, pp.680, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[8] T. Albrecht, A. Rossberg, J.D. Albrecht, J.P. Nicolay, B.K. Straub, T.S. Gerber, M. Albrecht, F. Brinkmann, A. Charbel, C. Schwab, and J. Schreck, "Deep Learning–Enabled Diagnosis of Liver Adenocarcinoma", *Gastroenterology*, vol. 165, no. 5, pp.1262-1275, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[9] M. Furuzuki, H. Lu, H. Kim, Y. Hirano, S. Mabu, M. Tanabe, and S. Kido, "A detection method for liver cancer region based on faster R-CNN", In *2019 19th International Conference on Control, Automation and Systems (ICCAS)*, pp. 2019, 808-811. [CrossRef] [Google Scholar] [Publisher Link]

[10] S. Lal, D. Das, K. Alabhya, A. Kanfade, A. Kumar, and J. Kini, "NucleiSegNet: Robust deep learning architecture for the nuclei segmentation of liver cancer histopathology images", *Computers in Biology and Medicine*, vol. 128, pp.104075, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[11] P. G. Sreelekshmi, P. Linu Babu and P. Josephin Shermila, "Leukemia classification using a fusion of transfer learning and support vector machine", *International Journal of Current Bio-Medical Engineering*, vol. 01, no.01, pp. 01-08, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[12] A. Prasanth, and N. Muthukumaran, "Primary open-angle glaucoma severity prediction using deep learning technique", *International Journal of Current Bio-Medical Engineering*, vol. 01, no.01, pp. 30-37, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[13] T. Thanjaivadivel, S. Jeeva, and A. Ahilan, "Real time violence detection framework for football stadium comprising of big data analysis and deep learning through bidirectional LSTM". 2019. [CrossRef] [Google Scholar] [Publisher Link]

[14] M. Chen, B. Zhang, W. Topatana, J. Cao, H. Zhu, S. Juengpanich, Q. Mao, H. Yu, and X. Cai, "Classification and mutation prediction based on histopathology H&E images in liver cancer using deep learning", *NPJ precision oncology*, vol. 4, no. 1, pp.14. 2020. [CrossRef] [Google Scholar] [Publisher Link]

[15] J. Amin, M.A. Anjum, M. Sharif, S. Kadry, A. Nadeem, and S.F. Ahmad, "Liver tumor localization based on YOLOv3 and 3D-semantic segmentation using deep neural networks", *Diagnostics*, vol. 12, no. 4, pp.823, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[16] D.T. Kushnure, and S.N. Talbar, "MS-UNet: A multi-scale UNet with feature recalibration approach for automatic liver and tumor segmentation in CT images", *Computerized Medical Imaging and Graphics*, vol. 89, pp.101885, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[17] V. Hemalatha, and C. Sundar, "Automatic liver cancer detection in abdominal liver images using soft optimization techniques", *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, pp.4765-4774, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[18] B. Sakthisaravanan, and R. Meenakshi, "OPBS-SSHC: outline preservation-based segmentation and search-based hybrid classification techniques for liver tumor detection",

Multimedia tools and applications, vol. 79, pp.22497-22523, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[19] X. Dong, Y. Zhou, L. Wang, J. Peng, Y. Lou, and Y. Fan, "Liver cancer detection using hybridized fully convolutional neural network based on deep learning framework", *IEEE Access*, vol. 8, pp.129889-129898, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[20] C.A. Hamm, C.J. Wang, L.J. Savic, M. Ferrante, I. Schobert, T. Schlachter, M. Lin, J.S. Duncan, J.C. Weinreb, J. Chapiro, and B. Letzen, "Deep learning for liver tumor diagnosis part I: development of a convolutional neural network classifier for multi-phasic MRI", *European radiology*, vol. 29, pp.3338-3347. 2019. [CrossRef] [Google Scholar] [Publisher Link]

[21] D. Anandan, S. Hariharan, and R. Sasikumar, "Deep learning based two-fold segmentation model for liver tumor detection", *Journal of Intelligent & Fuzzy Systems*, pp.1-16. [CrossRef] [Google Scholar] [Publisher Link]

[22] A. Nithya, A. Appathurai, N. Venkatadri, D.R. Ramji, and C.A. Palagan, "kidney disease detection and segmentation using artificial neural network and multi-kernel k-means clustering for ultrasound images", *Measurement*, vol. 149, pp.106952. 2020. [CrossRef] [Google Scholar] [Publisher Link]

[23] A. Jegatheesh, N. Kopperundevi and M. Anlin Sahaya Infant Tinu, "Brain aneurysm detection via firefly optimized spiking neural network," *International Journal of Current Bio-Medical Engineering*, vol. 01, no.01, pp. 23-29, 2023. [CrossRef] [Google Scholar] [Publisher Link]

AUTHORS

R. Ramani Associate Professor, Computer Science and Engineering in P.S.R. Engineering College, Sivakasi. She completed her Ph.D. in Information and Communication Engineering from Anna University in the year 2021. She obtained her M.E Computer Science and Engineering in the year 2007 from Annamalai University, Chidambaram and B.E Information Technology in the year 2004 from P.S.R Engineering College, Sivakasi. She has 15 years of teaching experience. Her research interests are Big Data Analytics, Data Science, Machine Learning and Cloud Computing. She has published more than 15 research articles in journals, conference proceedings. She has published 5 patents. She acts as reviewer in reputed journals. She delivered guest lecture in the topic of Big Data Analytics and Data Analyst. She holds membership in ACM, CSI Chapter and Life Member in ISTE.

K. Vimala Devi an active teacher and researcher and working as Associate Professor in the School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu, India. She received her PhD degree in the year 2008, and M.E in Computer Science and Engineering in the year 2003, from the Department of Computer Science and Engineering, College of Engineering, Guindy campus, Anna University, Chennai, India. She did her MCA in the year 1997 from University of Madras, Chennai. Having above 26 years of experience in teaching out of which 15 years in research, she has guided 10 research scholars for Ph.D. in the areas of Computer Networks, Wireless Networks, Network Security, Text mining, Image Processing, Software Testing, Cloud Computing and Big Data. Published about 120 research papers in International/National Journals and conferences, out of which 19 SCI journal papers indexed in ACM portal and 39 articles, indexed in Scopus and 15 IEEE conf. publications. Reviewer in IEEE Communications letter, IEEE Access, Elsevier Journal of Parallel and Distributed Computing, IJCS and IJNM of John Wiley publications.

P. Thiruselvan received his Bachelors of Engineering, Computer Science and Engineering from Arulmigu Kalasalingam College of Engineering in the year 2007 and Master of Engineering - Computer Science and Engineering in P.S.R Engineering College under Anna University in the year 2012. He is Pursuing Ph.D. in Anna University. Presently, he is working as an Assistant Professor in the Department of Computer Science and Engineering, P.S.R. Engineering College, Sivakasi. His area of interest includes Cyber Security, Big Data Analytics, Cloud Computing and Machine Learning. He has more than 10 years of experience in teaching and 3 years in industry as a web programmer. He has published more than 3 research articles in reputed international journals, conference proceedings. He has published 3 patents. He also holds professional membership in ISTE and CSI.

M. Uma Maheswari received her Bachelors of Engineering, Computer Science and Engineering from SCAD College of Engineering and Technology in the year 2010 and Master of Engineering - Computer Science and Engineering in P.S.R Engineering College under Anna University in the year 2023. Presently, she is working as an Assistant Professor in the Department of Computer Science and Engineering, P.S.R. Engineering College, Sivakasi. Her area of interest includes Machine Learning and Deep learning. She has 7 months of experience in teaching and 2 years in industry as a software programmer.

Arrived: 25.07. 2024
Accepted: 28. 08. 2024