Indian Journal of Pure & Applied Physics
Vol. 63, November 2025, pp. 1052-1065
DOI: 10.56042/ijpap.v63i11.23382

# Physics Inspired Optimisation and Explainable AI Framework for Enhanced BHP Flooding Attack Classification in Optical Burst Switching Networks

Arun Kumar S[a*], Sasikala S[a], Anusha K[a] & Gopinath P[b]

[a]Department of Electronics and Communication Engineering, Kumaraguru College of Technology, Coimbatore 641 049, India

[b]School of Electronics Engineering, Vellore Institute of Technology, Vellore 632 014, India

*Received: 2nd September 2025; accepted: 6th October 2025*

Optical Burst Switching (OBS) networks offer high bandwidth efficiency and low latency, making it an ideal choice for next generation high-speed photonic communications. However, the burst based transmission architecture is highly vulnerable to flooding attacks, which can severely degrade network performance. In this work, a hybrid approach for Burst Header Packet (BHP) flooding attack classification is proposed. The method combines direct Machine Learning (ML) on tabular data, tabular to image conversion using EfficientNet-b0 fine-tuning. Further, deep EfficientNet-b0 features are optimized using physics inspired Black Hole Optimisation with Adaptive Mutation (BHO-AM). Finally, the optimised features are classified using a Bayesian-optimized Support Vector Machine (SVM) classifier. Explainable AI (XAI) techniques, including Grad-Class Activation Map (CAM) and Occlusion Sensitivity, are employed to enhance interpretability and identify the most critical features influencing classification. Experimental results show that Efficient Net fine-tuning achieves 99.50 % accuracy, while the BHO-AM optimized SVM model attains 99.60 % accuracy with significantly reduced training time. This study introduces a novel tabular to image conversion framework for OBS attack data, enabling Deep Learning models to achieve high accuracy. Thus, combining XAI methods, with DL classification the proposed work achieves better performance and enhanced interpretability in OBS networks.

**Keywords:** Optical burst switching, Flooding attacks, Deep learning, Explainable AI, Feature selection

## 1 Introduction

The need for high-speed data transmission has made Optical Communication (OC) a fundamental technology in today's modern communication systems. OC utilizes light to transmit data through optical fibers, offering high bandwidth, long distance data transmission, faster transmission rates, and strong immunity to electromagnetic interference. These characteristics have positioned optical technologies as key components of optical networks, which serve as the backbone infrastructure for internet traffic, data centers, and cloud-based services[1].

Optical Burst Switching is a networking paradigm used in Optical Networks (ON) to manage the bursty and dynamic nature of internet traffic. OBS combines the advantages of circuit switched and packet switched optical networks by clustering data packets into larger burst before data transmission[2]. Each data burst contains a Burst Header Packet for storing the essential information of the burst. BHP is sent prior to data transmission to reserve network resources of the burst and configure switching elements in the intermediate node, thereby allowing bursts to traverse the network with a minimal delay[3].

BHPs traverse the control plane and are processed electronically, which makes them vulnerable to various security threats and attacks. Common BHP-related attacks include flooding attacks, burst header spoofing, Denial of Service (DoS) attacks, burst scheduling manipulation, and contention-based congestion exploitation. Among these, the BHP Flooding (BHPF) attack is a major concern in OBS networks. Such attacks can overwhelm the control plane, exhaust network resources, and result in denial of service to legitimate users. These attacks can significantly degrade network performance, restrict data flow, and even render services inaccessible, making attack classification a critical component of secure OBS network design.

Current advancements in Artificial Intelligence (AI) offer promising solutions for intelligent anomaly detection in optical networks. A survey on the use of AI methods for enhancing the attack classification accuracy in OBS networks is presented in this section.

Deep Learning (DL) architectures such as Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) are explored for mitigating

*Corresponding author: E-mail: arunkumar.s.ece@kct.ac.in

BHP attacks, with a maximum accuracy of 95.87% achieved using the CNN algorithm[3]. The Ant Colony Optimisation (ACO) algorithm is applied for feature selection on BHP attack data, followed by classification using SVM and Extreme Learning Machine (ELM), where SVM achieves 91% accuracy[4]. A semi-supervised K-Means algorithm is used for BHP attack classification, yielding an accuracy of 90.20%[5]. Variants of K- Nearest Neighbour (KNN) classifiers and standard ML classifiers are experimented for improving the classification performance in OBS networks. KNN with a third order distance function resulted in an improved accuracy of 99.30 %[6].A two-stage feature selection framework is analysed, employing the Fisher Score in the first stage for initial feature ranking. In the second stage, optimal features are selected using the Gorilla Troops Optimizer (GTO) algorithm and classified using an ELM classifier, achieving an accuracy of 91.00%[7]. A DL framework is employed for enhanced classification of BHPF attacks. After initial pre-processing, feature selection using Elephant Herd Optimisation (EHO) is employed to select discriminant features and further classified by Mobile Net CNN. An improved accuracy over 99.00 % is obtained[8]. A semi supervised learning with self-training is proposed for attack classification in OBS networks. An accuracy of 89.22 % is reported for classification of flooding attacks using modified self-training algorithm[9]. Information gain-based feature selection followed by decision tree classifier is experimented for enhanced classification of flooding attacks[10]. A baseline system for flooding attack classification is employed using conventional ML classifiers. Further to improve the performance, Particle Swarm Optimisation (PSO) is applied for feature selection. Optimized features classified by SVM Classifier resulted in an enhanced performance of 95.00 %[11].An attack detection framework in OBS networks using SVM and K instance-based classifier is proposed, and improved performance metrics are observed[12]. A semi supervised modified self-training methodology in conjunction with Extra Trees Classifier resulted in BHPF attack classification accuracy of above 98.00 %[13]. BHP data fed as input to customised 7layer CNN yielded an accuracy of 99.00 %[14].

AI models behave like a black box, making their decisions difficult to interpret. XAI addresses this challenge by translating opaque model outputs into interpretable predictions. Traffic management and attack classification in networks is enhanced by employing XAI techniques in conjunction with ML and DL. The use of XAI has reduced the computational overhead and improved the system performance[15]. To enhance the interpretability and transparency of link failures in optical fiber, XAI approach is proposed. Local and global explanations are used along with classification for better explanation and effective classification[16]. In the 6th Generation (6G) Communication system, network slicing is employed for resource management. XAI employed in network slicing improves reliability in real time decision making and risky environments. The role of XAI in improving resource utilization, fault identification and fairness in AI decision making makes it a critical choice for performance monitoring and improvement in 6G systems[17]. The dynamic routing problem in optical networks can be enhanced by adding XAI, which helps in understanding the transmission and reception parameters that most significantly influence performance. Thus, XAI enables better interpretation of ML model decisions and ranks the influential features. The insights provided by XAI support targeted optimisation to improve overall performance of the optical networks[18]. Therefore, the proposed work integrates DL with XAI to enhance attack classification accuracy and identifying the parameters most responsible for attack classification. This combination ensures dependable detection and greater trust in the model's results.

From the literature survey, most existing methods for detecting attacks in OBS networks use traditional ML on tabular data, which limits the accuracy to below 99%. DL methods like CNNs have not been widely used because the data is not in image form, and current studies rarely explain which features influence the classification.This gap highlights the need for a novel framework that transforms tabular data into image form, applies advanced DL for classification, and incorporates XAI to both enhance accuracy and improve interpretability.

The significant contributions and novelty of the proposed work are:

i This work experiments with a novel method for converting tabular OBS network traffic data into grayscale images, enabling the use of deep CNN models for classification.

Table 1 — Common Attacks in OBS Networks

| Attack Type | Description |
|---|---|
| BHP flooding | Overloads the control plane by transmitting multiple fake BHPs to reserve resources, causing legitimate bursts to be dropped. |
| Burst Spoofing | Sends BHPs with false source or destination addresses to mislead routing and waste resources. |
| Burst Scheduling manipulation | Exploiting weaknesses in burst scheduling algorithms to cause inefficient resource allocation or conflicts. |
| Denial of Service | Continuously utilize network resources to block access for legitimate users. |
| Contention manipulation | Generating burst contentions at core nodes to disrupt actual data transmission and degrade service. |

ii The proposed framework employs EfficientNet-b0 for fine-tuning and feature extraction, achieving high classification accuracy with efficient computation.
iii Metaheuristic feature selection is performed using the Black Hole Optimisation with Adaptive Mutation (BHO-AM) algorithm to remove irrelevant and redundant features.
iv SVM classifiers are optimized through Bayesian methods for automatic hyperparameter tuning and improved generalization.
v Explainable AI techniques, including Grad-CAM and Occlusion Sensitivity, are applied to identify and validate the most influential features contributing to classification decisions.

## 2 Methodology

### 2.1 Overview of Optical Burst Switching Networks

OBS is used as an advanced switching technique in next generation ON for better bursty traffic management and to handle the high bandwidth. OBS uses the merits of both circuit switching and packet switching to handle the dynamic traffic and resource allocation effectively.

An OBS network consists of Edge nodes, Core nodes and Control and data planes.

***2.1.1 Edge nodes:*** Edge nodes (Ingress/Egress nodes) groups packets into bursts at the receiver. A control packet called BHP is transmitted over the control channel for reserving wavelength and data along the transmission path.

***2.1.2 Core nodes:*** Core nodes are routers that operate in optical domain and facilitates burst data forwarding based on the BHP header information. Core nodes have limited buffering capability.

***2.1.3Control and data plane:*** Control plane uses BHP to handle scheduling and signalling of bursts. Data plane is responsible for actual data transmission.

OBS transmission consists of following steps:

i Data packets are aggregated into bursts at the Ingress nodes based on a criterion.
ii Transmission of BHP in the control channel prior to the data burst.
iii Processing of BHP electronically at core nodes to reserve resources.
iv Transmission of data bursts and BHP along the reserved path utilising the allocated resources.

The OBS architecture offers better flexibility and low latency but also exposes the control plane to security vulnerabilities and attacks. The various attacks in OBS networks are listed in Table 1.

BHP flooding attack is the most common and critical attacks in OBS networks. BHP targets the control plane, which is essential for managing burst data transmission. BHP flooding attack is hard to detect, increases control plane load, blocks legitimate traffic and degrades the overall Quality of Service (QoS) of the network. The steps involved in BHP flooding is illustrated in Fig. 1.

### 2.2 Machine Learning based pipeline for Proposed DL based Attack Classification

ML based approaches have been widely used for BHP flooding attacks as they can learn complex patterns in network behaviour. This enables accurate differentiation between normal and malicious BHP traffic. Furthermore, ML approaches can adapt to evolving attack strategies and minimize false alarms, enhancing their suitability for high-speed OBS network environments. Henceforth, the proposed approach uses ML in conjunction with optimisation algorithm-based feature selection for effective Classification of BHP Flooding attacks. Explainable AI based approaches are also used for enhancing the interpretability of the proposed attack classification framework. The proposed system architecture of the proposed DL-XAI and Physics inspired optimisation framework for OBS attack classification is presented in Fig. 2.

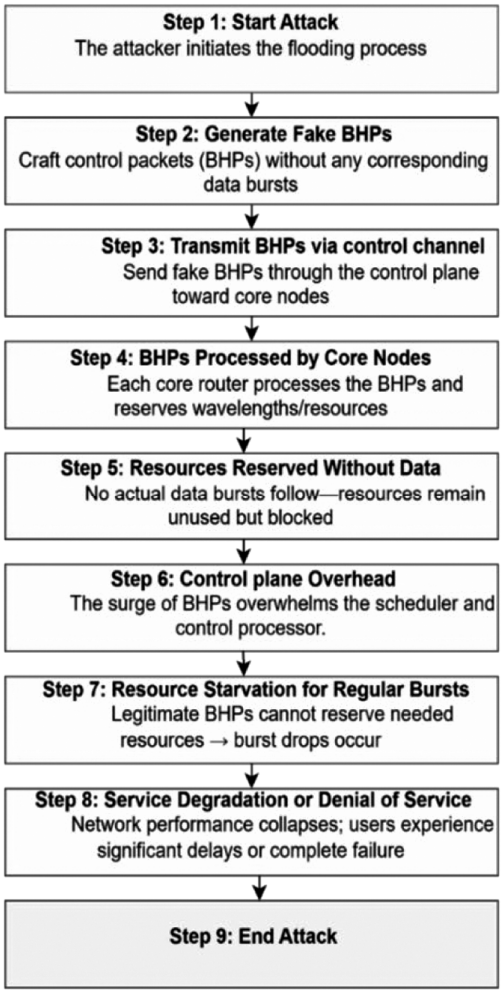


Fig. 1 — Steps involved in BHP Flooding Attack

#### *2.2.1 Dataset*

The dataset available in UCI repository is used in this study[19-21]. The dataset contains 21 features, and the feature description is presented in Table 2. The dataset for this study was generated with a network topology enabling flexible configuration of Optical Burst Switching (OBS) scenarios, including normal operation, contention, congestion, and flooding attacks. Initially, edge nodes were labelled as Behaving (B) or Misbehaving (M) to form a binary dataset for detecting improper resource reservations. This was later extended to a multi-class dataset with four labels No Block, M-No Block, M-Wait, and Block to capture diverse BHP flooding conditions. Simulations varied bandwidth (100 Mbps to 1 Gbps) and attacker positions to generate diverse traffic patterns. Data was pre-processed by averaging variables over 10 iterations, and class labels were assigned based on expert analysis of false resource utilization and packet drop rates for accurate classification.

#### *2.2.2 Data Pre-processing*

In the pre-processing stage, missing attributes in the data are removed. Synthetic Minority Oversampling Technique (SMOTE) is utilized to avoid class imbalance in the attack dataset. Each class is expanded to contain about 500, resulting in a balanced dataset of 2006 instances across four categories. The class distribution before and after SMOTE is presented in Table 3. This ensures fair

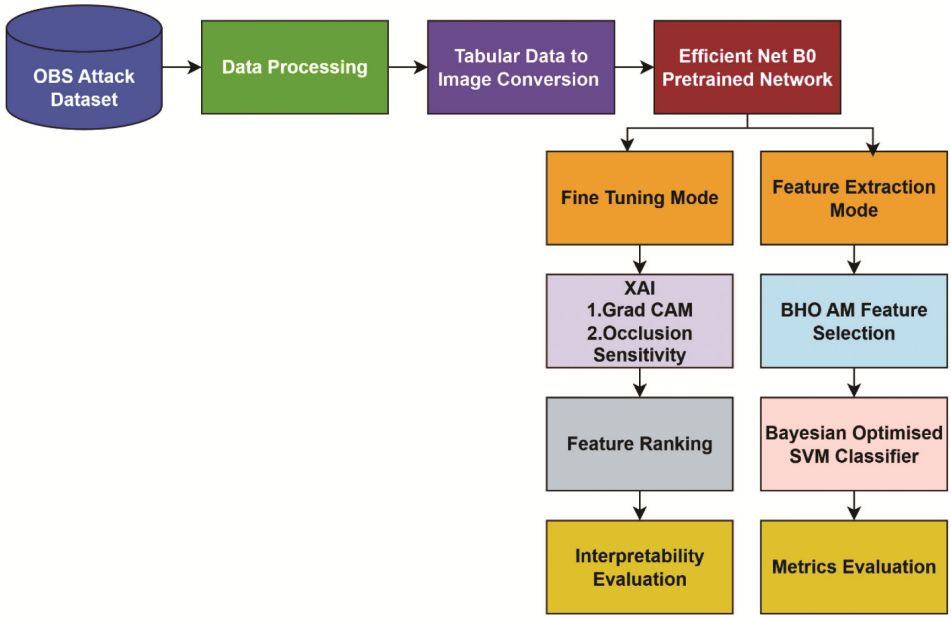


Fig. 2 — Proposed System Architecture for enhanced BHP flooding attack Classification

Table 2 — Description of the attributes used in the dataset

| Feature | Attribute | Description |
|---|---|---|
| F1 | Source Node Identifier | Identifier of the node that initiates data transmission. |
| F2 | Rate of Bandwidth Utilization | Normalized value of bandwidth that is eligible for reservation. |
| F3 | Ratio of Packet Loss | Ratio of packets lost to the total packets sent per node. |
| F4 | Allocated Bandwidth Capacity | Initial bandwidth allocation assigned to each node by the user. |
| F5 | Mean Delay per Second | Average latency experienced at each node, measured per second. |
| F6 | Lost Packet Percentage | Percentage of total packets lost at each node. |
| F7 | Lost Byte Percentage | Byte-level data loss percentage per node. |
| F8 | Packet Reception Rate | Number of data packets successfully received per second on the reserved bandwidth. |
| F9 | Bandwidth Consumption | Actual bandwidth consumed by each node within the allocated limit (as per F4). |
| F10 | Bandwidth Loss | Unutilized or wasted bandwidth within the allocated bandwidth (F4) at each node. |
| F11 | Packet Size (in Bytes) | Size of each data packet transmitted by a node, measured in bytes. |
| F12 | Number of Packets Sent | Total count of data packets sent per second from a node over its allocated bandwidth. |
| F13 | Number of Packets Received | Total number of packets accepted by each node every second. |
| F14 | Number of Packets Dropped | Total count of packets that were not delivered successfully, per node, per second. |
| F15 | Bytes Transmitted | Volume of data (in bytes) sent by each node per second. |
| F16 | Bytes Received | Volume of data (in bytes) received per second by a node over the reserved bandwidth. |
| F17 | Average Packet Drop (across 10 runs) | Average packet loss rate (from F3) calculated over 10 simulation iterations. |
| F18 | Average Bandwidth Usage (over 10 runs) | Average bandwidth usage (from D9) recorded across 10 simulation cycles. |
| F19 | Average Delay (over 10 runs) | Mean delay per node, obtained over 10 runs of the simulation. |
| F20 | Behavioural node Status | Classification of node behavior as normal, abnormal, or potentially abnormal. |
| F21 | Flooding Rate Indicator | Measured flooding intensity or rate observed at each node. |
| Label | Class Label | Label defining node condition: Block, No Block (NB), NB-No Block, or NB-Wait. |

Table 3 — Class distribution Before and After SMOTE

| Class | Before SMOTE | After SMOTE |
|---|---|---|
| Block | 120 | 500 |
| NB-NoBlock | 500 | 504 |
| NB-Wait | 300 | 501 |
| No Block | 154 | 501 |

training for classification models and improves their ability to detect minority class anomalies. Since the node status attribute (F20) in the dataset is categorical, One-Hot Encoding (OHE) is applied to convert it into a numerical format resulting in 23 features.

#### *2.2.3 Image generation*

To leverage Deep CNNs (DCNNs) for tabular numeric features, a transformation from numerical data to image data is performed. In this approach, each row (instance) in the dataset, is normalized to numerical feature values ranging from 0 to 1 and is reshaped into a fixed-size gray scale image. Since CNNs operate on 2D inputs, the 23-dimensional feature vector is zero-padded to 25 and reshaped into a 5×5 matrix, where 5×5 is chosen to balance feature representation and compatibility with convolutional filters. Each pixel in the image represents one normalized feature, and its brightness encodes the feature's relative magnitude - where 1 corresponds to maximum intensity (white) and 0 to minimum intensity (black). This transformation captures spatial patterns among features, enabling CNN-based models to extract deep spatial representations for effective classification.

The sample images of Block, NB-No Block, NB-wait and No block classes of OBS dataset is presented in Fig. 3.

#### *2.2.4 Efficient Net B0 Pretrained Network*

Efficient Net is a family of DCNN technique that optimises the parameters of the model (resolution, width and depth) simultaneously[22]. EfficientNet-b0 is a lightweight pretrained model that employs squeeze and excitation networks and mobile inverted bottleneck convolution to achieve high accuracy with minimal parameters. Efficient Net is chosen over other pretrained networks because it achieves a better balance between the model accuracy and computational cost through its compound scaling

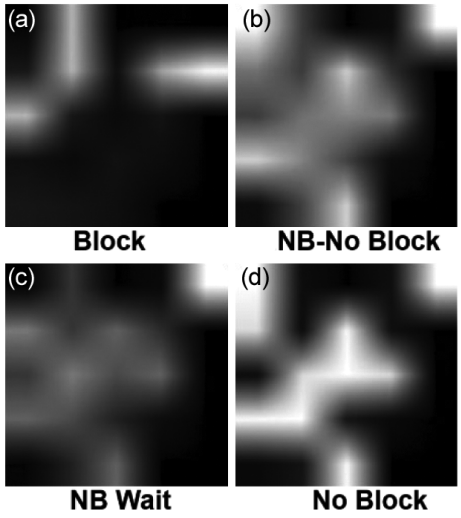


Fig. 3 — Sample Images of BHP attacks

strategy, which optimally scales the network parameters. Furthermore, lightweight architecture makes Efficient Netb0 suitable for fast training and inference while maintaining high classification performance. In this work, Efficient Netb0 is used in two modes: fine-tuning for end-to-end training and as a deep feature extractor for subsequent classification.

#### *2.2.5 Feature Extraction*

The transformed image in the image generation step is passed through Efficient Netb0 to generate high-level feature maps from the Global Pooling layer. These feature representations capture both spatial patterns and hierarchical structures in the input, providing an efficient yet informative summary of each sample in a lower-dimensional space. The extracted features are then used as inputs for ML classifiers to perform classification of BHP flooding attacks in OBS networks.

#### *2.2.6 Feature Selection*

Physics based meta heuristic feature selection is employed for selecting discriminating features. Black Hole Optimisation (BHO) is a metaheuristic inspired algorithm modelled based on the astronomical behaviour of black holes in the space[23-24]. In BHO, the candidate solutions are modelled as stars, and the best candidate is selected as black hole. The movement of the stars toward the black hole in the search space create a gravitational pull. Stars that move within a defined distance, known as the event horizon, are considered absorbed by the black hole. To maintain population diversity, these stars are replaced with newly generated random solutions.

```
Input:
  - Fitness function f(x)
  - Population size N
  - Dimensions D (features)
  - Maximum iterations T
Output:
  - Best solution BH and its fitness
1. Initialize population X with N random stars of D dimensions
2. Evaluate fitness of all stars
3. Identify the star with the best fitness → set as Black Hole (BH)
4. For t = 1 to T:
   a. Compute adaptive mutation probability:
      P_mut = 0.5 - 0.45 * (t / T)
   b. For each star Xi ≠ BH:
      i. Move Xi towards BH:
         Xi = Xi + rand * (BH - Xi)
      ii. Apply adaptive mutation:
         For each dimension d = 1 to D:
            If rand < P_mut:
               Xi[d] = 1 - Xi[d]   // Bit flip
      iii. Evaluate new fitness of Xi:
         If f(Xi_new) < f(Xi_old):  // better than itself
            Update Xi
         If f(Xi_new) < f(BH):      // better than current BH
            BH = Xi_new             // update black hole
   c. Calculate event horizon:
      R = f(BH) / sum(fitness of all stars)
   d. For each Xi ≠ BH:
      If distance(Xi, BH) < R:
         Replace Xi with new random solution
5. Return BH and its fitness
```

Fig. 4 — Pseudo code of BHO AM

A novel optimisation algorithm called “BHO with Adaptive mutation (BHO-AM)” is experimented in this study. BHO-AM outperforms standard BHO by introducing a dynamic mutation mechanism that enhances exploration and avoid premature convergence by improving the exploitation-exploration balance. This adaptive strategy maintains population diversity and improves the chances of selecting more relevant features, leading to better optimisation results in complex search spaces.

The steps in BHO – AM optimisation algorithm illustrated as pseudocode in Fig. 4. In feature selection, BHO treats the candidate solutions (stars) as binary vector with selected features. The star that exhibits better classification performance is considered as Black Hole (BH) and other stars move towards BH to inherit its characteristics. An adaptive mutation mechanism is applied to each bit to maintain diversity and avoid local optima problems. In addition, a unique mechanism called Event Horizon replaces the stars that are similar to BH with new randomly generated solutions to preserve population diversity. The iterative process in BHO-AM, results in

selection of discriminative features for better classification.

BHO-AM is applied as a wrapper-based feature selector, where candidate feature subsets are evaluated using classification accuracy as the guiding criterion. Each solution is represented as a binary vector, with '1' for selected features and '0' for unselected ones. The objective function combines classification error and the proportion of selected features as:

$$f_{k} = \rho\, E + (1-\rho)\frac{\sum_i k_i}{N}. \quad .. (1)$$

where $f_k$ denotes the Fitness function corresponding to solution vector $k$ of length $N$, where entries are binary (1 = feature selected, 0 = feature not selected). $N$– Total number of available features in the dataset.

E – Classification error obtained from the chosen subset of features.$\rho$ – Weighting parameter that balances classification performance and subset size, typically set to 1 to emphasize classification accuracy.

#### *2.2.7 Classification*

In this study, SVM classifier with linear and nonlinear kernel function is used for classifying the attack types. SVM is a maximum margin classifier that aims to find the hyperplane which separates different classes in a high dimensional space. SVM is also effective in handling nonlinear problems using kernel transformation. Bayesian Optimisation (BO) is used for tuning hyper parameters of SVM model for enhanced performance.

### 2.3 Explainable AI based pipeline for Proposed DL based Attack Classification

XAI provides transparency in detecting BHP flooding attacks in OBS networks by revealing the influence of individual features on model predictions. It ranks network parameters according to their importance, helping to validate and trust model decisions. XAI also assists in identifying the most relevant features, improving classification efficiency. Overall, it ensures both interpretability and reliability in high-speed optical networks.

Fine-tuned EfficientNet-b0 model is used for attack classification. The model predicts the class labels from the input images. XAI based framework is employed to identify the regions in the image that influence the model's decision[25-26]. Grad-CAM and Occlusion sensitivity XAI based techniques are employed in this work for better interpretation. Grad-CAM highlights the regions in an image that contribute most to a model's decision by using the gradients of the target class flowing into the final convolutional layer[27]. Occlusion sensitivity measures the change in prediction confidence when specific regions of the input are masked, helping identify areas most influential to the model's output[28].

The steps involved in XAI framework are as follows:

#### *2.3.1 Grad CAM Implementation*

i Compute the prediction score for each class, by passing the input image through the EfficientNet-b0 model.
ii Activation maps are created from the final convolution layer of EfficientNet-b0 pretrained model.
iii Gradients of the predicted class score w.r.t each channel in the activation maps are computed using back propagation algorithm.
iv Spatial averaging is done on the gradients to obtain the importance weight for each channel
v A class specific attention heat map is generated from the weighted sum of activation maps.
vi Rectified Linear Unit (ReLU) operation is applied to retain the positive contributions w.r.t the specified class.
vii The heat map is overlaid on the original image for better visualization

#### *2.3.2 Occlusion Sensitivity Implementation*

i Partition images into overlapping patches using sliding window mechanism
ii For each patch position, replace the corresponding region in the image with a predefined occlusion mask
iii Pass the occluded image through the trained model and measure the class confidence score of the target class
iv Calculate the score difference by computing the change in the predicted class confidence score relative to the original image for the corresponding patch location
v Assign the score differences back to corresponding patch positions to generate a relevance map, where high values indicate significant region for prediction.
vi Normalize the relevance map to [0,1] and visualised as colour overlay on the original image.

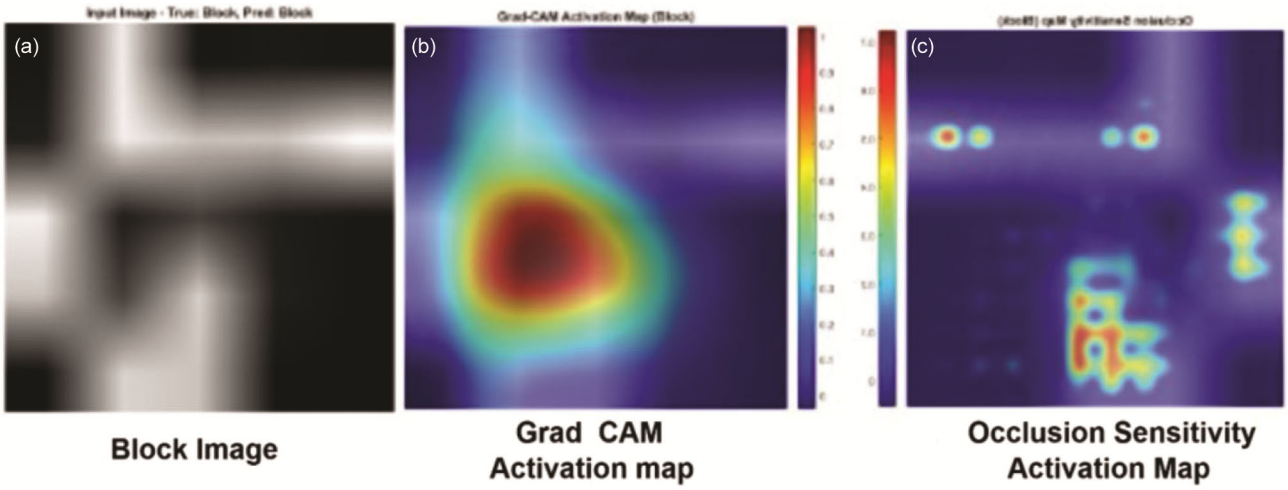


Fig. 5 — Activation Maps of Grad CAM and Occlusion Sensitivity correponding to Block class

Sample images of Grad-CAM and Occlusion sensitivity activation maps for Block class is shown in Fig. 5.

#### *2.3.3 Class level Averaging*

For the target class, all images correctly classified by EfficientNet-b0 model is used to generate Grad-CAM and Occlusion sensitivity heat map. A pixel wise averaging is performed on the heat maps to generate class level Grad CAM/ Occlusion sensitivity heat map. This averaging ensures consistent patterns of prominent regions per class.

#### *2.3.4 Feature Space Mapping*

The averaged heat maps are resized to 5 x 5, as the original images are obtained from 25 feature values (after padding) from the tabular data. From the resized heat maps, intensity value at each position is considered as important score for the corresponding feature.

#### *2.3.5 Feature ranking*

Top 9 features are ranked based on the highest intensity important score, indicating their prominence in feature classification.

## 3 Results and Discussion

### 3.1 ML Based Analysis

The metrics used for ML based analysis is specified in Table 4. A Baseline system is constructed by directly classifying the attacks using SVM classifier. The results are tabulated in Table 5. An accuracy of 70.47 % is achieved using linear SVM classifier. Cubic kernel SVM achieved the highest classification performance with 96.42% accuracy, 96.05% sensitivity, and 98.51% specificity, indicating strong discriminative capability. The Gaussian kernel showed comparable but slightly lower results, while the Quadratic kernel yielded moderate performance. Performance drop in linear kernel, signifies that the dataset's decision boundaries are highly nonlinear, favouring nonlinear kernel functions for optimal classification.

Table 4 — Evaluation metrics

| Metric (%) | Equation | |
|---|---|---|
| Accuracy | $ACC = \frac{\alpha + \delta}{\alpha + \beta + \gamma + \delta}$ | ... (2) |
| Sensitivity | $SEN = \frac{\alpha}{\alpha + \gamma}$ | ... (3) |
| Specificity | $SPE = \frac{\delta}{\beta + \delta}$ | ... (4) |
| Precision | $PRE = \frac{\alpha}{\alpha + \beta}$ | ... (5) |
| F1Score | $F1 = 2 \times \frac{PRE \times SEN}{PRE + SEN}$ | ... (6) |
| Matthews Correlation Coefficient | $MCC = \frac{\alpha.\delta - \beta.\gamma}{\sqrt{(\alpha+\beta)(\alpha+\gamma)(\delta+\beta)(\delta+\gamma)}}$ | ... (7) |

True/False Positive - (α/β), True/False Negative - (δ/γ)

To further enhance the performance, Transfer learning using EfficientNet-b0 Pretrained network is experimented. Pretrained EfficientNet-b0 acts as backbone and the final three layers of the model (drop out, fully connected and classification) are replaced with parameters of OBS dataset. Grayscale images are converted to colour images and resized to 224×224 to match the input size of Efficient Net model. Adam optimiser is used for model training with a learning rate of 0.0001 for 6 epochs and a mini batch size of 16. Validation accuracy of 99.50 % is observed. Accuracy and loss plot for Efficient Net model is shown in Fig. 6. The training time for the model is 2.33 hours.

Table 5 — BHP attack classification using ML classifiers

| Algorithm | ACC (%) | SEN (%) | SPE (%) | PRE (%) | F1 (%) | MCC (%) |
|---|---|---|---|---|---|---|
| Linear | 70.47 | 69.26 | 88.36 | 70.97 | 69.88 | 58.65 |
| Cubic | 96.42 | 96.05 | 98.51 | 97.56 | 96.75 | 95.35 |
| Gaussian | 95.19 | 93.86 | 98.10 | 95.74 | 94.67 | 92.90 |
| Quadratic | 87.08 | 87.47 | 94.87 | 88.92 | 88.03 | 83.03 |

Table 6 — BHP attack classification using Efficient Netb0

| ACC (%) | SEN (%) | SPE (%) | PRE (%) | F1 (%) | MCC (%) |
|---|---|---|---|---|---|
| 99.50 | 99.50 | 99.83 | 99.51 | 99.50 | 99.34 |

Table 7 — BHP attack classification using Efficient Net deep features and ML Classifiers

| SVM Type | ACC (%) | SEN (%) | SPE (%) | PRE (%) | F1 (%) | MCC (%) |
|---|---|---|---|---|---|---|
| Linear | 93.02 | 93.04 | 97.67 | 93.06 | 92.94 | 90.72 |
| Cubic | 97.71 | 97.70 | 99.23 | 97.87 | 97.74 | 97.01 |
| Gaussian | 96.56 | 96.55 | 98.85 | 96.97 | 96.63 | 95.60 |
| Quadratic | 97.36 | 97.37 | 99.12 | 97.39 | 97.34 | 96.50 |
| Optimised | 98.60 | 98.61 | 99.53 | 98.61 | 98.60 | 98.14 |

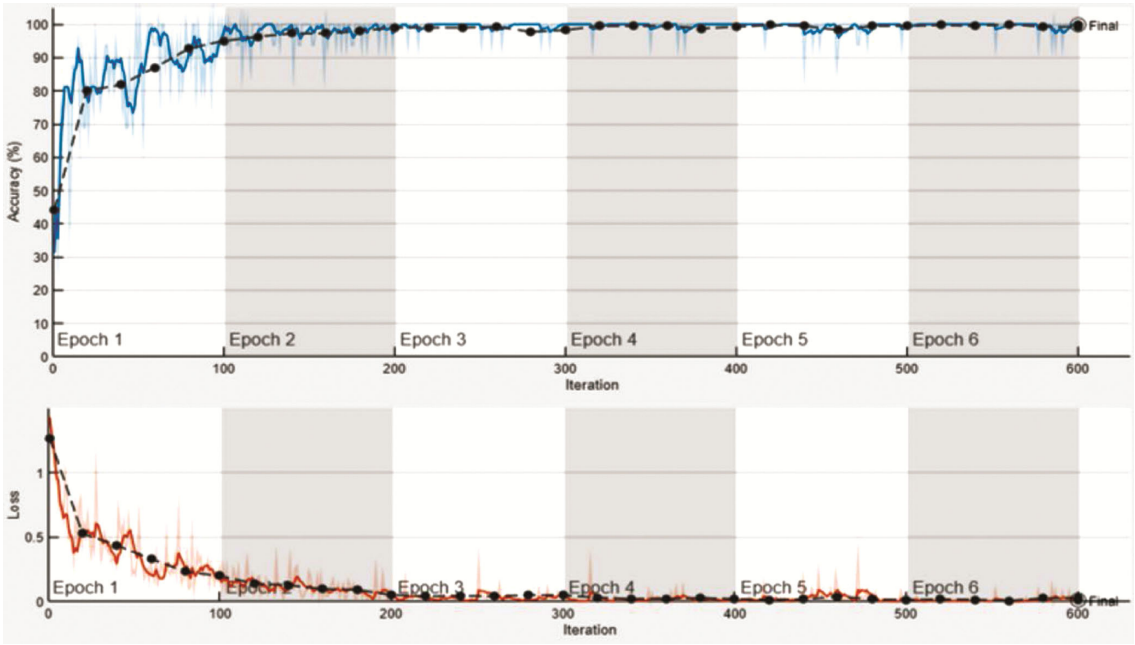


Fig. 6 — Accuracy and loss plot for EfficientNet-b0

The metrics of pretrained EfficientNet-b0 model are tabulated in Table 6. The fine-tuned EfficientNet-b0 model achieved near-perfect performance, with 99.50% accuracy, high sensitivity, precision, and F1- score, along with very high specificity (99.83%) and MCC (99.34%). These results indicate better generalization and strong discriminative ability of the model.

Raw grayscale images contain fine details difficult to capture by shallow neural networks. DCNN trained on ImageNet data set extract discriminative and hierarchical feature maps for effective attack classification. Therefore, to further achieve training time and accuracy trade-off, features extracted from the activations of the global average pooling layer of Efficient Net model are classified by SVM algorithm. Final pooling layer consists of rich sematic information for better classification. 10-fold cross validation is employed for reducing bias and over fitting. The results of Efficient Net deep features classified by SVM and its variants is tabulated in Table 7. Classification of EfficientNet-b0 deep features using SVM showed high performance across all kernel types, with cubic and quadratic kernels performing better than linear. Bayesian-optimized SVM achieved the highest results, with 98.60% accuracy and

Table 8 — BHP attack classification using BHO AM optimised Efficient Net deep features and ML classifiers

| SVM Type | ACC (%) | SEN (%) | SPE (%) | PRE (%) | F1 (%) | MCC (%) |
|---|---|---|---|---|---|---|
| Linear | 91.82 | 91.84 | 97.28 | 91.85 | 91.71 | 98.12 |
| Cubic | 98.45 | 98.46 | 99.49 | 98.46 | 98.45 | 97.94 |
| Gaussian | 96.81 | 96.80 | 98.93 | 97.17 | 96.86 | 95.91 |
| Quadratic | 97.26 | 97.27 | 99.09 | 97.31 | 97.23 | 96.37 |
| Optimised | 99.60 | 99.60 | 99.87 | 99.61 | 99.60 | 99.47 |

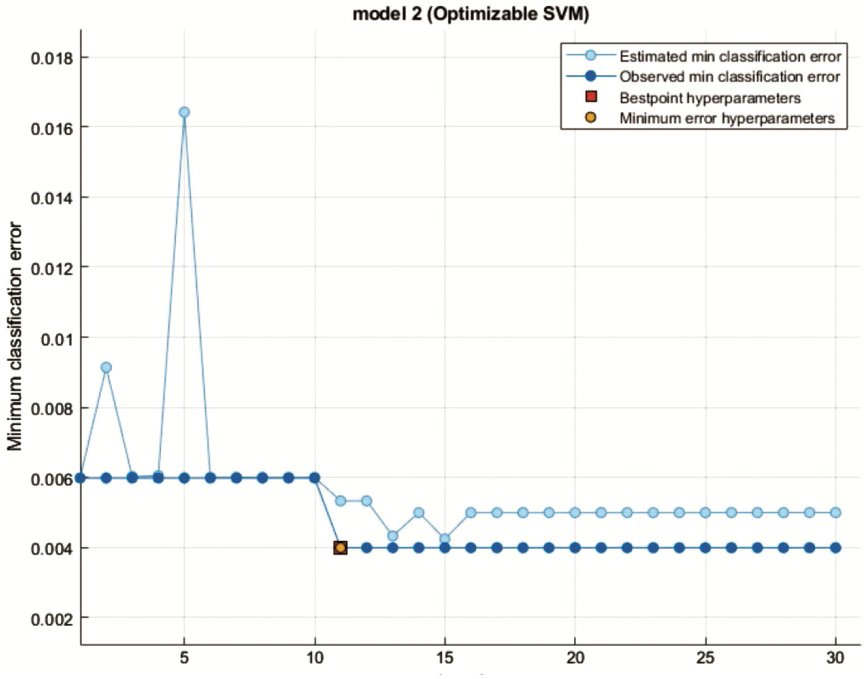


Fig. 7 — MCE plot of BHO AM optimised Efficient Net deep features classified by BO SVM

MCC of 98.14%, reflecting optimal balance between sensitivity and specificity. These results highlight the effectiveness of combining deep features with optimized ML classifiers. To further reduce the feature dimension of deep features extracted from EfficientNet-b0 model, a meta heuristic optimisation-based feature selection using novel BHO-AM optimisation algorithm is experimented.

Following the feature selection, feature dimension is halved to 640 from 1280. Despite the reduced feature set, classifier resulted in better performance using SVM classifier. The use of Adaptive mutation, preserves population diversity and avoids local minima, resulting in optimal feature subset. A comparable performance metrics also retained w.r.t. original full feature set. Accuracy of 99.60 % indicate that the BHO-AM based feature selection reduced the feature space and enhanced the classifier's ability to select relevant discriminative features, making it a better approach for high-dimensional deep features obtained from pretrained CNNs. Bayesian optimised SVM classifier is used for attack classification. Linear kernel function with box constraint level of 228.8474, kernel scale of 1 and one vs one classification resulted in accuracy of 99.60 % using Bayesian optimisation. The results of optimised deep features classified by different SVM kernels are presented in Table 8. Classification using EfficientNet-b0 features selected via BHO-AM demonstrated strong performance across all SVM kernels. The cubic kernel performed exceptionally well (98.45% accuracy), while the Bayesian-optimized SVM achieved the highest metrics with 99.60% accuracy, 99.87% specificity, and MCC of 99.47. These results indicate that feature selection further improves classification efficiency while maintaining high predictive accuracy. Minimum Classification Error (MCE) plot of BHO AM optimised Efficient Net deep features classified by BO SVM is shown in Fig. 7. Confusion matrix in terms of False Negative rates (FNR) and False Discovery rates (FDR) is presented in Figs. 8 - 9 respectively. Receiver Operating Characteristic

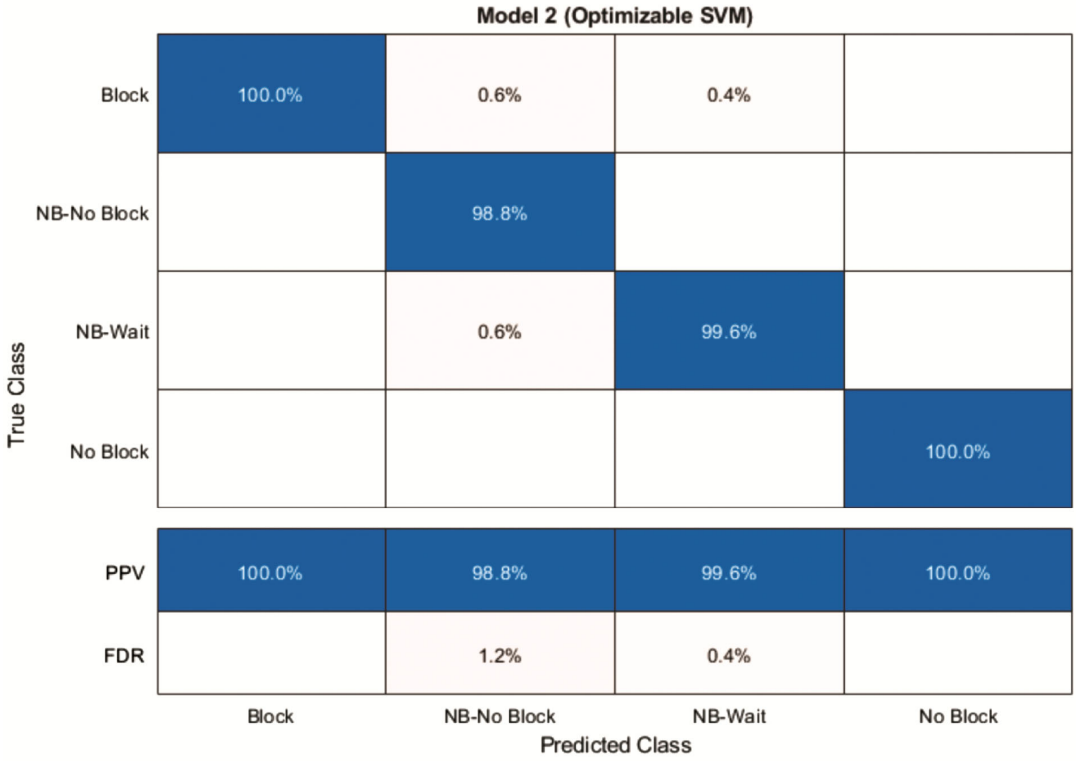


Fig. 8 — Confusion matrix in terms of FNR (BHO AM+BO SVM)

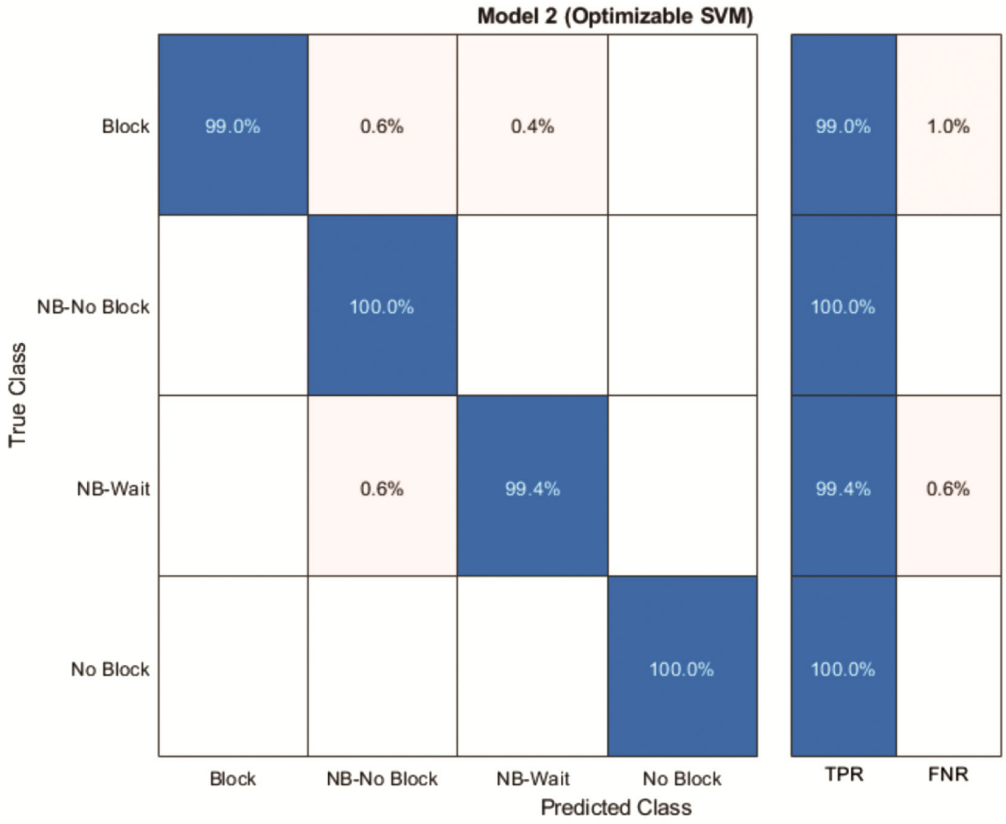


Fig. 9 — Confusion Matrix in terms of FDR (BHO AM+BO SVM)

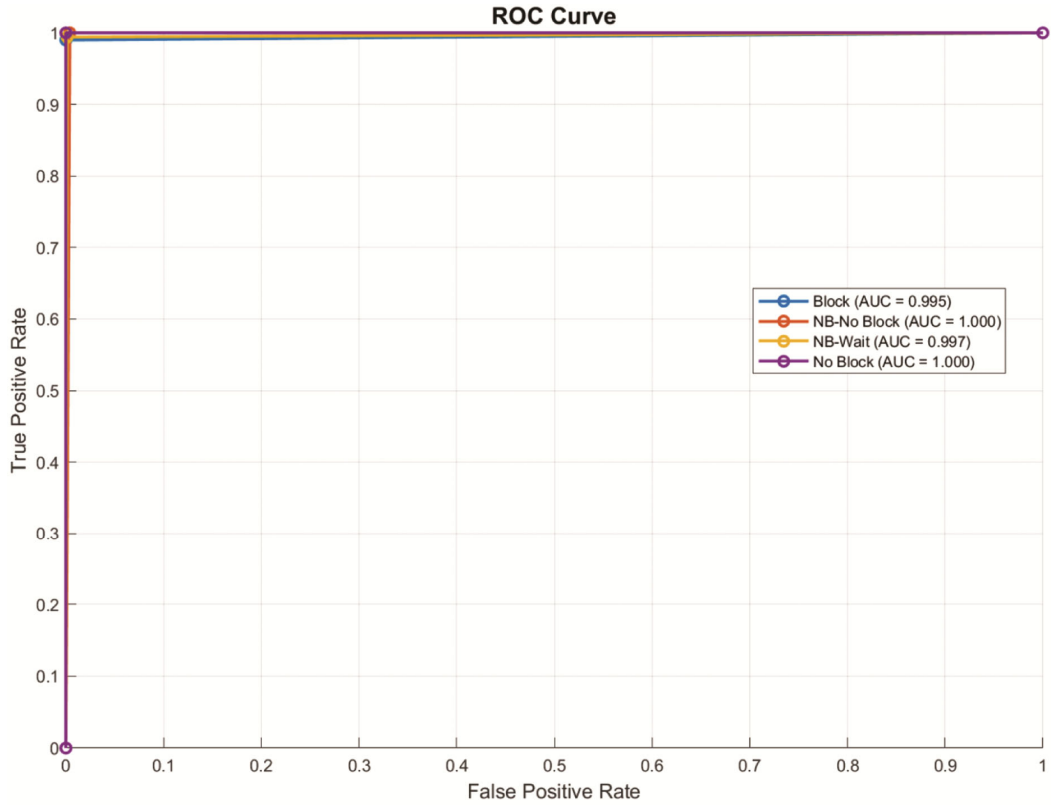


Fig. 10 — ROC AUC curve (BHO AM+BO SVM)

(ROC) – Area under curve (AUC) is presented in Fig. 10. ROC AUC is close to 1 for all classes, indicates the better discriminative capability of classifier and high overall performance of the model in terms of sensitivity and specificity.

The end-to-end fine-tuned EfficientNet-b0 resulted in highest single-model accuracy of 99.50%, when the network is directly adapted for the classification task. The alternative pipeline of extracting deep features from Efficient Net and classifying with an SVM after feature selection via BHO-AM achieved comparably high performance of 99.60% accuracy**,** while substantially reducing the feature dimensionality. This indicates that compact, optimized deep feature representations can approach the accuracy of fully fine-tuned networks with reduced training time. The fine-tuned Efficient Netb0, while more accurate (99.50%), has a higher computational complexity, and consumes a training time of 2.33 hours. In contrast, the Efficient Net + BHO-AM + optimized SVM reduces complexity, achieving 99.60% in 1.5 hour.

### 3.2 XAI Based Analysis

The proposed Explainable AI pipeline using Grad-CAM and Occlusion Sensitivity provides better interpretation into the decision-making process of the fine-tuned Efficient Net-b0 model. Both visualization techniques highlight concentrated regions corresponding to a small subset of features that had the highest influence on classification. By mapping the class level averaged heat maps back to the original 5×5 feature space, the framework identified the top nine most influential features for each class. The best features obtained via Grad CAM and Occlusion Sensitivity is shown in Figs. 11-12. Combining XAI with classification improves trust in the deep learning model and aids in understanding feature relevance for attack classification. The Grad-CAM analysis identifies Packet Reception Rate (F8), Bytes Received (F16), Flooding rate indicator (F21), Average Packet drop (across 10 runs) (F17), and Lost Byte Percentage (F7) as the most influential features driving Efficient Netb0's classification decisions. These features primarily reflect real-time traffic flow, loss characteristics, and abnormal traffic patterns, which are highly relevant in detecting network attacks.

Occlusion Sensitivity reveals a slightly different ranking, with Bandwidth Loss (F10), Number of Packets dropped (F14) and Bandwidth Consumption (F9) emerging as top contributors. These metrics

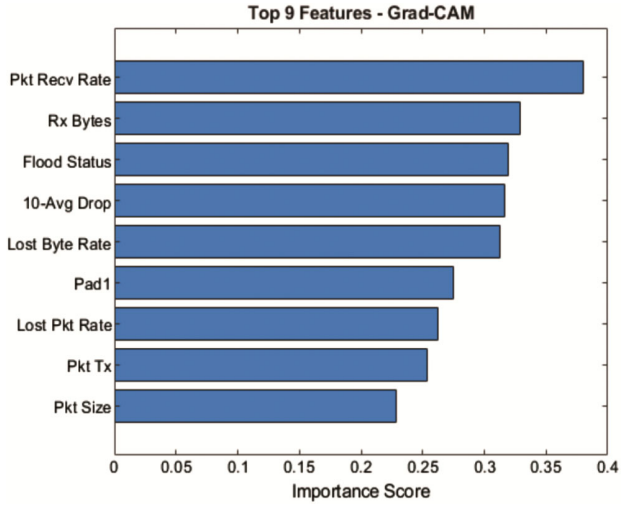


Fig. 11 — Top 9 features selected by Grad CAM

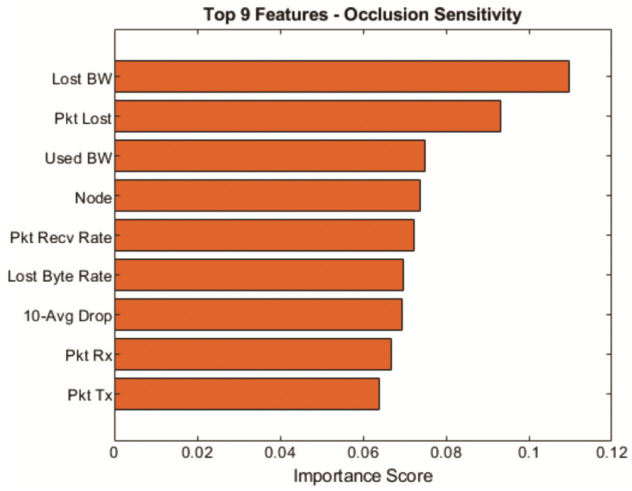


Fig. 12 — Top 9 features selected by Occlusion Sensitivity

emphasize bandwidth inefficiencies and direct packet loss, highlighting regions where performance degradation is most impactful for classification.

A partial overlap is observed between the two XAI methods for example, Packet Reception Rate (F8) and Lost Byte Percentage (F7) are consistently ranked highly in both approaches. This correlation strengthens confidence in their importance as robust indicators for attack classification. However, differences in the rankings reflect the complementary nature of Grad-CAM (gradient based feature importance) and Occlusion Sensitivity (perturbation based importance), suggesting that the model leverages both instantaneous traffic behaviour and simulated perturbation responses to make accurate predictions.

The explain ability analysis from Efficient Netb0's feature selection shows that both Grad CAM and Occlusion Sensitivity highlight key traffic and loss-related metrics as critical for detecting network attacks. Grad CAM emphasizes features related to real-time flow and anomaly detection, while Occlusion Sensitivity focuses more on bandwidth efficiency and packet loss. The differences in ranking illustrate that combining gradient-based and perturbation-based explanations provides a comprehensive understanding of the model's decision-making process. Thus, combining XAI with DL enhances the model performance and improves interpretability for better decision making.

### 3.3 Discussion

The performance comparison of the proposed work with existing works in literature is presented in Table 9. Performance comparison clearly shows that

Table 9 — Performance Comparison of proposed work with Existing Literature

| Reference | Classifier | Key Highlights | Accuracy (%) |
|---|---|---|---|
| [6] | Modified KNN | Multiple ML algorithms are experimented with BHP attack classification | 99.30 |
| [7] | ELM classifier | Two stage Feature selection using fisher score and GTO followed by ELM classifier | 91.00 |
| [8] | Deep CNN | EHO algorithm for feature selection and Mobile Net for classification | 99.27 |
| [9] | Semi Supervised ML (SSML) | Four different Semi supervised ML algorithms for classification of BHP attacks. Modified Self-training model resulted in high metrics. | 89.22 |
| [11] | SVM | PSO optimized features classified by SVM | 95.00 |
| [13] | SSML | Semi supervised learning with Extra Tree classifier | 99.00 |
| [21] | Decision Tree (DT) Classifier | Statistical feature selection followed by rule-based DT classifier | 87.00 |
| [14] | Deep CNN | Features fed as input to 7 layered custom CNN for enhanced performance | 99.00 |
| Proposed work | Deep CNN+XAI | Tabular data is transformed to images and classified by Efficient Netb0 pretrained network.<br>Efficient Net deep features classified by BHO-AM feature selection and Bayesian optimized SVM Classifier.<br>Grad CAM and Occlusion sensitivity based XAI for better interpretation | 99.50<br>99.60 |

the proposed work achieves the highest accuracy of 99.5%, surpassing both deep CNN-based and traditional machine learning approaches reported in previous studies. Previous works relied solely on handcrafted features or deep learning without feature selection. In contrast, the proposed method integrates tabular-to-image transformation, EfficientNet-b0 in dual modes, and BHO-AM feature selection to capture richer and more discriminative patterns, achieving a maximum accuracy of 99.6%. Furthermore, Bayesian optimized SVM ensures optimal classification performance, and the use of Grad-CAM and Occlusion Sensitivity adds interpretability, which is absent in other methods. Thus, the proposed approach leads to improved performance and transparency in BHPF attack detection.

## 4 Conclusion

This work presents an effective framework for classifying flooding attacks in OBS networks through a novel tabular to image conversion approach and DL via Efficient Net fine-tuning. A metaheuristic feature selection using BHO-AM integrated with a Bayesian-optimized SVM is employed for improved accuracy and efficiency. The proposed method achieves up to 99.50% accuracy with deep fine-tuning and 99.60% accuracy with optimized features, significantly reducing training time while maintaining equivalent performance. Furthermore, XAI techniques such as Grad-CAM and Occlusion Sensitivity provide valuable insights into the most influential features, enhancing interpretability. Future work will focus on validating the method across multiple OBS datasets and real-time traffic scenarios, as well as investigating advanced feature fusion and ensemble-based XAI techniques to further enhance both performance and interpretability.

## References


1 He J, Norwood R A, Brandt-Pearce M, Djordjevic I B, Cvijetic M, Subramaniam S, Himmelhuber R, Reynolds C, Blanche P, Lynn B & Peyghambarian N, *Comput Electr Eng*, 40 (1) (2014) 216.

2 Tiwari G, Chauhan R C & Ratnesh R K, *Opt Quantum Electron*, 57 (4) (2025) 1.

3 Mohammad Q S, Zaman A S & Bhutto N M, *TuijinJishu / J Propuls Technol*, 45 (3) (2025) 2024.

4 Takieddine Seddik M, Kadri O, Bouarouguene C & Brahimi H, *ComputSist,* 25 (2) (2021) 423.

5 Hossain M K H, *Baghdad Sci J*, 16 (3) (2019) 36.

6 Nuha H H, Mugitama S A, Absa A A & Sutiyo, *IoT*, 6 (1) (2024) 1.

7 Benyahia A, Kadri O & Moumen, *Social Sci Res Netw*, 1 (2025) 1.

8 Vahalingam R, Rajagopal B & Arumugam S, *Informacije MIDEM* , 53 (3) (2023) 167.

9 Hossain M K, Haque M M & Dewan M A, *Computers*, 10 (8) (2021) 95.

10 Almaslukh B, *Secur Commun Netw*, (1) (2020) 8840058.

11 Liu S, Liao X & Shi H, *Photonics*, 8 (12) (2021) 555.

12 Efeoğlu E &Tuna G, *Balkan J Electr Comput Eng*, 9 (4) (2021) 342.

13 Hossain M K & Haque M M, *Int J Electr Comput Eng*, 10 (4) (2020) 4340.

14 Hasan M Z, Hasan K Z & Sattar A, *Procedia Comput Sci*, 143 (2018) 970.

15 Zilli C, Sacco A, Esposito F & Marchetto G, *IEEE Trans Netw Serv Manag*, 22 (4) (2025) 3617.

16 Theodorou G, Karagiorgou S, Fulignoli A & Magri R, *World Conference on Explainable Artificial Intelligence*, (2024) 268.

17 Sun H, Liu Y, Al-Tahmeesschi A, Nag A, Moghadam S, Canberk B, Arslan H & Ahmadi H, *IEEE Open J Commun Soc*, 6 (2025) 1372.

18 Goścień R, *IFIP Networking Conference*, (2024) 750.

19 Rajab A, *Int J Sci Appl Inf Technol*, 8 (2019) 164.

20 Rajab A, Huang C T, Al-Shargabi M & Cobb J, *International Conference on Information Security Practice and Experience*, (2016) 315.

21 Rajab A, Huang C T & Al-Shargabi, M, *Opt Switch Netw*, 29 (2018) 15.

22 Tan M & Le Q, *International conference on machine learning,* (2019) 6105.

23 Hatamlou A, *Inf Sci*, 222 (2013) 175.

24 Abualigah L, Elaziz M A, Sumari P, Khasawneh A M, Alshinwan M, Mirjalili S, Shehab M, Abuaddous H Y & Gandomi A H, *Appl Intell* 52 (10) (2022) 11892.

25 Soothar K K, Chen Y, Memon K A, Magsi A H, Khan A & Qureshi K K, *Arab J Sci Eng*, (2025) 1.

26 Nazim S, Alam M M, Rizvi S S, Mustapha J C, Hussain S S & Suud M M , *PLOS One*, 20 (5) (2025) 1.

27 Wang S & Zhang Y, *Comput Mater Contin*, 76 (2) (2023) 1321.

28 Valois P H, Niinuma K & Fukui K *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision,* (2024) 4829.