FMDB *Transactions on Sustainable Computer Letters*

Crime Detection Using Lyrebird Optimization Algorithm-Based AdaHybridANet Classification

K. Aravinda[1,*], V. Revathi[2], Thirumalraj Karthikeyan[3], S. Venkatasubramanian[4], Mykhailo Paslavskyi[5]

[1]Department of Electronics and Communication Engineering, New Horizon College of Engineering, Bengaluru, Karnataka, India.
[2]Department of Research and Development, New Horizon College of Engineering, Bengaluru, Karnataka, India.
[3]Department of Artificial Intelligence, Trichy Research Labs, Quest Technologies, Tiruchirappalli, Tamil Nadu, India.
[4]Department of Computer Science and Business Systems, Saranathan College of Engineering, Trichy, Tamil Nadu, India.
[5]Department of Computer Science, Ukrainian National Forestry University, Lviv, Lviv Oblast, Ukraine.
aravindake@gmail.com[1], revshank153@gmail.com[2], thirumalraj.k@gmail.com[3], veeyes@saranathan.ac.in[4], mykhailo.paslavskyi@nltu.edu.ua[5]

Abstract: Crime rates are rising worldwide, making automated crime detection systems more vital. This study uses Fast Local Laplacian Filter FLLF preprocessing to improve the UCF-Crime dataset photos for automated crime identification. This solution preserves sharp edges and visual features needed to identify crimes. A novel interactive attention strategy is employed to extract features from the Visual Transformer (ViT) and Res2Net, collectively referred to as ViTRes2DualNet, leveraging their combined capabilities. The Adaptive Hybrid Attention Network (AdaHybridANet) is used for categorisation to maximise crime-detection accuracy. The proposed classifier enhances feature extraction by integrating the Coordinate Attention and Enhanced Non-Local Attention (ENLA) modules to extract both local and global features. The suggested model's hyperparameter tuning utilises the Lyrebird Optimisation Algorithm (LBOA) to simulate Lyrebirds in danger. LBOA replicates escape and hiding by mathematically modelling Lyrebirds' threat scanning and decision-making. Results revealed 99.32% accuracy for the suggested strategy. When utilising innovative AdaHybridANet classification and LBOA tuning, the proposed method outperforms existing methods. With less information wasted, this approach delivers high-quality images for more accurate detection.

Keywords: Contrast Enhancement; Fast Local Laplacian Filter; Convolutional Neural Network; Vision Transformer; Lyrebird Optimisation Algorithm; Enhanced Non-Local Attention; Adaptive Hybrid Attention Network.

Received on: 03/01/2025, **Revised on:** 12/03/2025, **Accepted on:** 25/04/2025, **Published on:** 22/11/2025

Journal Homepage: https://www.fmdbpub.com/user/journals/details/FTSCL

DOI: https://doi.org/10.69888/FTSCL.2025.000485

Cite as: K. Aravinda, V. Revathi, T. Karthikeyan, S. Venkatasubramanian, and M. Paslavskyi, "Crime Detection Using Lyrebird Optimization Algorithm-Based AdaHybridANet Classification," *FMDB Transactions on Sustainable Computer Letters*, vol. 3, no. 4, pp. 183–197, 2025.

1. Introduction

Crime detection plays a vital role in maintaining law and order and ensuring the safety of society. Communities develop and grow in response to the complexity and variety of criminal activity [1]. The importance of efficient crime detection methods is

*Corresponding author.

growing. It has never been easy for law enforcement organisations around the world to detect and prevent illegal activity, ranging from minor thefts to serious crimes [2]. Historically, human intervention and observation have been the primary methods used for crime detection. However, automated crime detection services have gained popularity due to technological advancements [3]. By leveraging a range of tools and techniques, including computer vision, machine learning, and artificial intelligence, these technological advancements enhance the efficacy and precision of crime detection [4]. Specifically, Convolutional Neural Networks (CNNs) have transformed the field of crime detection by automatically analysing images and videos associated with criminal activity [5]. A deep learning architecture known as a CNN was developed specifically to process visual data efficiently. CNNs are very good at extracting relevant information, intricate patterns, and features from images, which is useful for spotting suspicious activity or objects in crime detection [6]; [7]. These networks scan image or video frames to recognise objects, detect irregularities, and pinpoint specific actions that may indicate criminal activity [8]. CNNs are trained on datasets of images or videos of criminal activity to detect it.

The networks can differentiate between suspicious and normal behaviour by recognising distinctive visual cues or patterns associated with criminal activity [9]. Because they have been trained on extensive datasets containing a variety of examples of criminal activity, they can recognise unique visual cues associated with various types of crimes. In addition to other criminal behaviours, they can be trained to recognise weapons, violent acts, theft, and vandalism [10]; [11]. By automating the monitoring and analysis of security footage, they remove the need for manual review and enable quicker response times to potential threats. Law enforcement can detect suspicious activity with the aid of CNNs, given their high accuracy and speed in handling large volumes of visual data [12]. These networks possess the flexibility to continually improve their detection capabilities, progressively increasing their accuracy as they are continually exposed to new and diverse datasets [13].

Preprocessing is performed using the FLLF technique for contrast enhancement, and an innovative hybrid network is created to efficiently blend the benefits of Res2Net and the transformer, enabling the extraction of multiple features from valuable data and boosting classification performance. An integrated module with multiple layers is suggested to incorporate information with multiple features. This module combines the two extraction functions from the Res2Net and ViT branches to enhance representation. The two attention modules that comprise the suggested Adaptive Hybrid Attention Network (AdaHybridANet) are Coordinate Attention and Enhanced Non-Local Attention (ENLA). Furthermore, to effectively fuse features from both local and global levels, an Adaptive Feature Aggregation (AFA) module is proposed. LBOA is presented as an imitation of Lyrebirds in their native habitat. The primary source of inspiration for LOA comes from the tactics Lyrebirds use to avoid danger. Two phases comprise the mathematical modelling of LBOA theory: (i) exploration from the modelling of a plan of escape, and (ii) from the simulation of an exploitative strategy. To evaluate overall results, this paper quantifies performance metrics, including accuracy (ACC), specificity (SP), F1-score (F1), recall (RC), and precision (PR).

2. Related Works

YOLOv5 object detection, combined with Deep Sort, was the method recommended by Nazir et al. [14] for tracking individuals in a video; the resulting bounding box coordinates were then utilised as temporal features. The extracted temporal features were then used to model a time-series classification problem. The proposed method was measured and evaluated using the popular UCF Crime dataset against the state-of-the-art robust temporal feature magnitude (RTFM) method, which relied on the Inflated 3D ConvNet (I3D) preprocessing method. The results demonstrated an 8.45-fold increase in detection inference speed over the state-of-the-art RTFM, and an impressive 92% F1 score, outperforming RTFM by 3%. In addition, these results were obtained using this technique without requiring expensive data augmentation or image feature extraction. The study by Muneer et al. [15] features a large benchmark dataset comprising 900 cases, divided into two groups: 450 shoplifting cases and 450 non-shoplifting cases. The five shoplifting techniques were used to annotate the cases manually. Additionally, a method for detecting shoplifting was proposed to evaluate the generated dataset. The baseline techniques for assessing the dataset were 2D and 3D CNN. The suggested method, which combined Inception V3 and BILSTM, outperformed all baseline techniques with an accuracy of 81%. To support various applications related to human activity identification, such as movement, theft, and behaviour detection (e.g., robbery), the generated dataset was made publicly available [30].

The study by Mujahid et al. [16] utilised two highly successful models that were combined: CNN (GoogleNet) and LSTM. GoogLeNet was the most sophisticated architecture for gesture recognition and hand detection. By preserving temporal data, LSTM helped to avoid information loss. Throughout the testing and training phase, these two got along better. To identify different categories of self-efficacy, this study employed a multi-class classification approach based on the current state of hand movements, utilising visual data processing and feature extraction [31]. From a collection of images taken in various settings, including those involving human subjects, the recommended architecture extracted high-quality frames. After these frames were processed, features related to their hand and body orientations were taken out and examined. Four different efficacy-related classes, such as confident, cooperative, confused, and uncomfortable, were distinguished based on these characteristics. The features were extracted using a combination of short-term memory (LSTM) for classification and specialised CNN layers for feature extraction. The remarkable outcomes of this study showed that deep learning techniques can effectively recognise

human body gestures with 90.48% accuracy. Lye et al. [17] presented a novel approach for recognising routine activities from pre-segmented video clips. The pre-trained convolutional neural network (CNN) model, VGG16, was used to extract visual features from sampled video frames, and the recommended pooling scheme was then employed to combine these features. The solution combined appearance and motion features from video frames and optical flow images, respectively. The mean and max temporal pyramid (TPMM) and mean and max spatial pooling (MMSP) pooling techniques were recommended for producing the final video descriptor.

The feature was used to train a linear support vector machine (SVM) to recognise the different activities observed in the video clip. The proposed method was evaluated on three publicly available benchmark datasets. Studies were conducted to demonstrate the benefits of integrating motion and appearance features for recognising everyday activities. The results showed that the proposed method for identifying activities of daily living has potential for improvement. The proposed MMSP–TPMM method performed better than other methods across three public datasets: 96.11% on the FPPA dataset, 75.37% on the ADL dataset, and 90.68% on the LENA dataset for average per-class precision (AP) and accuracy, respectively. The image processing or deep learning algorithms employed in the work by Singla and Chadha [18] processed videos captured by electronic devices, such as cameras, in real time, saving significant time and labour. The Inception Model, using the SGD Optimiser and Leaky ReLU Activation Function, produced an accuracy of 83.43%, while the Ensemble Model yielded the highest accuracy at 86.6%. As a result, using decision-making, anomalies were successfully discovered in real-time surveillance scenarios. The study by Gupta et al. [19] initially focused on efficiently extracting discriminative representations for CNNs (e.g., C3D and I3D) and ViTs (e.g., CLIP) using two pre-trained feature extractors. Next, it used a proposed temporal self-attention network (TSAN) to consider both short- and long-term temporal dependencies and display engaging video clips. CNN-ViT-TSAN is a generalised architecture based on multiple instance learning (MIL) that was designed to specify a set of models for the WVAED problem by utilising TSAN and features derived from CNNs and/or ViTs.

The efficacy of the CNN-ViT-TSAN methodology was demonstrated through experimental results on extensively used public crowd datasets. Two DL-based schemes were combined in the study by Jeong et al [20]: The 3D Convolutional AutoEncoder (3D-AE) for anomaly detection and the SlowFast neural network for anomaly classification. The 3D-AE could locate anomalous event locations using regions of interest (ROIs) and the points generated within them. The SlowFast model could classify anomalous events based on the ROI. These multimodal strategies maximised the benefits of the security system while exacerbating its drawbacks. An attempt was made to enhance the efficacy of anomaly learning by utilising the virtual world of Grand Theft Auto 5 (GTA5) to generate a new dataset. The dataset comprised 78 normal-state data points and 400 abnormal-state data points, with clip sizes ranging from 8 to 20 seconds. A virtual data collection method was employed to supplement the original dataset, as it was challenging to replicate abnormal states in the real world. Consequently, the proposed method yielded an 85% classification accuracy, which is superior to the 77.5% accuracy achieved by a single classification model. Moreover, the trained model was validated against the GTA dataset using a real-world assault class dataset with 1300 replicated instances. As a result, 1100 attack data were successfully classified, yielding an accuracy of 83.5%. This demonstrated even more that the recommended strategy might provide outstanding results in real-world situations.

2.1. Problem Statement

Globally rising crime rates have made automated crime detection technologies more necessary. Often, current methods fail to preserve crucial image details necessary for accurate crime identification. To address this challenge, this work proposes a novel approach that enhances images from the UCF-Crime dataset via the Fast Local Laplacian Filter (FLLF) preprocessing technique. This method prioritises maintaining sharp edges and critical image details that are essential for criminal identification. ViTRes2DualNet, a blend of Res2Net and the Visual Transformer (ViT), is presented as an efficient feature-extraction method. Additionally, the method utilises the Adaptive Hybrid Attention Network (AdaHybridANet) for classification to enhance crime detection accuracy. By integrating the Coordinate Attention and Enhanced Non-Local Attention (ENLA) modules, the proposed classifier maximizes feature extraction by capturing both local and global features. Furthermore, by hyperparameter tuning with the Lyrebird Optimisation Algorithm (LBOA), it mimics Lyrebirds' hiding and evading behaviours in threat situations. The stated accuracy of 99.32% indicates the effectiveness of the recommended method, particularly when LBOA tuning and the state-of-the-art AdaHybridANet classification are employed. Ultimately, this strategy yields high-quality images, minimizes information loss, and significantly enhances the accuracy of crime detection compared to existing techniques.

3. Proposed Methodology

The steps involved in implementing the recommended strategy are shown in the schematic in Figure 1. This section covers the following procedures: image preprocessing using FLLF, extraction using ViTRes2DualNet, and classification using an LBOA-based AdaHybridANet classifier, which uses LBOA for hyperparameter tuning.

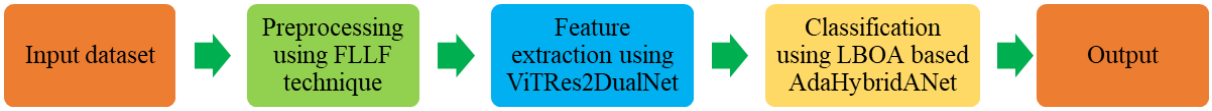

Figure 1: The proposed model's workflow

3.1. Dataset Description

The well-known UCF Crime dataset, which comprises two types of data—normal and criminal videos—and approximately 128 hours of video, was used in this study [21]. The videos that comprised the crime-labelled samples showed various criminal acts, including abuse, arrests, arson, assaults, car crashes, explosions, burglaries, fights, robberies, shootings, theft, shoplifting, and vandalism. Because shoplifting crimes often involve suspicious behaviour before the theft, this was taken into account in the study. Nonetheless, the suggested approach can be effectively employed in numerous video-based anomaly identification scenarios where an anomaly is defined as the way certain objects move.

3.2. Preprocessing using FLLF

Improving the quality of crime images, particularly for crime detection, was the primary objective of this study. The primary objective was to improve low-contrast, noisy images that make it hard to identify crimes with precision. The goal was to enhance the clarity and cleanliness of these photos to facilitate more effective and reliable crime detection. The FLLF technique is a sophisticated method for boosting contrast in photos without sacrificing their intricate details and textures. In contrast to conventional global contrast enhancement techniques, which may result in oversaturation or loss of fine details, the FLLF method operates locally, adapting to the specific characteristics of different image regions [22]. The Laplacian pyramid is the fundamental component of the FLLF methodology. To build this pyramid, a sequence of Gaussian pyramids at varying scales is created, and the differences between levels are computed. At the level of a Gaussian pyramid, mathematically, i is denoted as Gi, the matching Laplacian pyramid level equation (1) calculates Li as:

$$\mathrm{Li} = \mathrm{Gi} - \mathrm{resize}(\mathrm{Gi} + 1) \quad (1)$$

Wherein to resize $(Gi + 1)$ symbolises the level of the Gaussian pyramid $Gi + 1$ scaled to the measurements of $\mathrm{G}i$. During this procedure, a pyramid is created to record information about local images and their variations at various scales. The modified Laplacian is introduced by the FLLF technique(ML), acquired by applying For every level of the Laplacian pyramid, a bilateral filter. Two primary factors control the bilateral filter: the range's similarity and spatial distance, d, r. It enhances contrast while preserving edges and reducing noise. In terms of mathematics, the modified Laplace ML at level i can be communicated in equation (2) as:

$$\mathrm{MLi} = \mathrm{bilateralFilter}\ (\mathrm{Li}, \mathrm{d}, \mathrm{r}) \quad (2)$$

Where the lateral filter is located $(\mathrm{Li}, \mathrm{d}, \mathrm{r})$ reaches the Laplacian pyramid level by applying the bilateral filter Li with parameters d and r. Using the modified Laplacian transform, reconstruct the enhanced image ML from the base image. G_0. Equation (3) computes the image that was reconstructed, E as:

$$\mathrm{E} = \mathrm{G}_0 + \mathrm{resize}(\mathrm{ML}_0) + \sum \mathrm{i} = \mathrm{nResize}(\mathrm{MLi}) \quad (3)$$

Where n is the number of levels in a pyramid. Within this formula, G_0 is the original image; enlarge(ML_0) is the scaled-down equivalent of the altered Laplacian at the summit of the pyramid, and $\sum$nResize(MLi) is the total of all the modified Laplacian levels that have been resized throughout the pyramid. The technique can enhance distinct crime images by adjusting contrast locally, based on their individual features. More information can be found by dissecting equation (3):

- **E:** This is the improved image that will be produced; it will have more details and contrast.
- **G_0This represents the base image, typically the first image** that requires enhancement.
- **resize(ML_0):** At the summit of the pyramid, the modified Laplacian (MLi) is scaled to fit the base image's dimensions (G_0). To do this, make sure that the (MLi) and G_0 possess the same height and width.
- **$\sum$i = nResize(MLi):** The sum $(\sum)$ is assuming control of every level (i) of the pyramid Laplacian from 1 to n '. The matching modified Laplacian for every level (MLi) is scaled to fit the base image's dimensions. (G_0) and subsequently included in the outcome.

Thus, this formula indicates that the improved picture ' E ' is created by combining three elements:

- The initial base picture G_0.
- In the Laplacian pyramid, the modified Laplacian is the highest level, $(\text{resize}(ML_0))$, making sure it fits the proportions of G_0.
- The total of each pyramid level's modified Laplacian levels ($\sum i = \text{nResize}(MLi)$), where every level has been scaled to correspond G_0.

Through this process, an enhanced image that adjusts contrast locally is produced by combining the base image with details captured at different scales and local image variations from the Laplacian pyramid. The Laplacian pyramid, as well as the modified Laplacian with reduction, is utilised to enhance handling across various scales, thereby improving contrast and preserving fine details. The outcome, ' E ', represents the enhanced image while maintaining local characteristics and with better contrast. The FLLF method offers an advanced approach to enhancing photo contrast. By utilising the bilateral filter and the Laplacian pyramid, it is possible to achieve localised enhancement while preserving important textures and details. The FLLF technique's adaptability enables it to be applied to a wide range of images with varying textures and contrast levels, thereby improving visual quality without compromising the image's natural appearance. To enhance the accuracy of crime detection from captured images, FLLF was used to preprocess the crime images. The resulting preprocessed output was then fed into the extraction process.

3.3. Feature Extraction using ViTRes2DualNet

A two-branch architecture is proposed for extracting crime photos. Here, two effective feature extractors, ViT and Res2Net, are built using the methods described in the studies [23] and [24].

3.3.1. ViT Branch

As shown in Figure 2, A single stack of the same layer, each layer having two sub-layers, is combined with a multi-layer perceptron (MLP) section and a multi-head self-attention (MSA) module. Normalisation processing using layer normalisation (LN) occurs before the input image is fed into each sublayer. Using the residual connection, the acquired image is directly fused with the inputs following each sublayer. Ultimately, following the L network coding layers, the initial component of the sequence x_L^0 is sent to the MLP-composed category header to forecast the image's category y. The variable in between x_l' and the output x_l of the l-th layer are expressed in equations (4) and (5) as follows

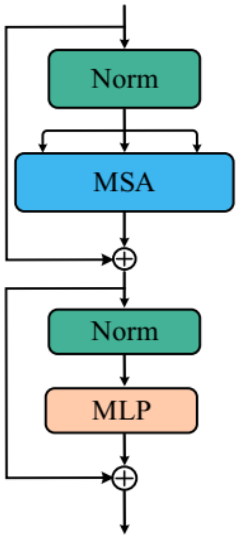

Figure 2: ViT encoder

$$x_l' = \text{MSA}\left(\text{LN}(x_{l-1})\right) + x_{l-1}, l = 1,2,\dots,L \quad (4)$$

$$x_l = \text{MLP}\left(\text{LN}(x_l')\right) + x_l', l = 1,2,\dots,L \quad (5)$$

Equation (6) can be used to get the final output.

$$y = \text{LN}(x_L^0) \quad (6)$$

3.3.2. Res2Net Branch

To obtain global characteristics and finer-grained details, Res2Net has been employed as a feature extractor. The feature maps in the Res2Net network are divided into four categories. $(p_i, i = 1,2,3,4)$ from the 1x1 convolution of the input layer [25]. The initial team p_1 is not processed, and a 3×3 convolution operation is applied to the other groups $(\kappa_i, i = 2,3,4)$. The final feature map adds the third and fourth groups. Equation (7) allows for the expression of the operations as mentioned above:

$$q_i = \begin{cases} p_i & i = 1 \\ \kappa_i(p_i) & i = 2 \\ \kappa_i(p_i + q_{i-1}) & i = 3,4 \end{cases} \quad (7)$$

In the channel dimension, the feature maps are combined. The output is then obtained via a 1x1 convolution. In the branch, a four-block feature extractor is built using Res2Net-50. To identify criminal activity more precisely from photos, the feature extraction procedure is followed by a classification step.

3.4. Classification using AdaHybridANet

The T1 and T2 MRI image stack is utilised by the proposed network, AdaHybridANet, to operate. Due to DenseNet-169's strong capabilities for feature extraction and propagation, it was employed [26]. Extracting both global and local features is essential to bolstering and supporting the architecture. As a result, the Coordinate Attention module and ENLA were utilised in the proposed work to extract both local and global salient features successfully. To better model notable correlations, the AFA module employs a squeeze-and-excitation mechanism to fuse features from surrounding layers adaptively. The AFA module's output feature map is then fed into a classifier block that includes flattening, linear classification layers, and global average pooling (GAP).

3.4.1. ENLA Module

Global information can be extracted from input images by introducing a non-local attention module. The ENLA module can capture long-term ties created by non-local activities. Additionally, contextual data is collected to enhance the model's ability to represent pixels in sequence. Equation (8) defines the non-local operation given a feature map x as input.

$$y_i = \frac{1}{H(x)} \sum_{\forall_j} f(x_i, x_j) g(x_j) \quad (8)$$

Where x and y are the non-local attention block's input and output, i is the output position's index, and j is the index for every position that needs to be computed. Equation (9) defines the normalisation factor, or H.

$$H(x) = \sum_{\forall_j} f(x_i, x_j) \quad (9)$$

Assuming a feature map input x_i, the feature map produced y_i is computed along the dimension using a softmax function j. The correlation function $f(x_i, x_j)$ is employed to calculate the degree of similarity as specified by equation (10). The role $g(x_j)$ calculates the input signal's representation at position j.

$$f(x_i, x_j) = \theta(x_i)^T \delta(x_j) \quad (10)$$

where $\theta(\cdot)$ and $\delta(\cdot)$ are feature transformations. Here, $\theta(x_i) = W_\theta x_j$ and $\delta(x_j) = W_\delta x_j$ are utilised to compute the input representation through linear embeddings. In practice, it is computed via matrix multiplication with a 1×1 convolutional layer. To further enhance feature propagation throughout the module and compute channel-wise attention, a residual link featuring softmax layers and average pooling is added. To extract salient features, this mechanism adaptively recalibrates and weights the channel information. The channel attention mechanism operates as a layer-by-layer normalisation of the channel information, facilitated by the numerous filters from previous levels in the DenseNet architecture. Furthermore, after the average pooling component generates attention vectors, the softmax layer computes the attention coefficients.

3.4.2. Coordinate Attention Module

While the ENLA block utilises the input images to identify prominent global features, the Coordinate Attention module captures specific local features. Additionally, this module was used in the research to maintain positional data, which is essential for characterizing spatial patterns. First, the mean values of the x- and y-axes are computed for every feature map channel using

global average pooling. Two pooling kernel spatial extents are used to accomplish this: (H,1) and (1, W), to encrypt every channel at the appropriate vertical and horizontal coordinates. The formulas for this are (11) and (12).

$$z_c^h(h) = \frac{1}{W}\sum_{0\le i\le W} x_c(h, i) \quad (11)$$

$$z_c^w(w) = \frac{1}{H}\sum_{0\le j\le H} x_c(j, w) \quad (12)$$

Direction-aware feature maps are generated by combining data from the horizontal and vertical directions using the two equations mentioned above. By preserving positional data along spatial directions, the attention module can locate features more precisely through these transformations. Equation (13) illustrates how the concatenated and transmitted combined features are used to reduce the number of channels through a pointwise convolutional layer.

$$f = \delta\left(F_1\left(\left[z^h, z^w\right]\right)\right) \quad (13)$$

Where $[\cdot,\cdot]$ denotes joining along a spatial dimension, δ is a non-linear activation function, and $f \in \mathbb{R}^{C/r\times h}$ This is the important spatial data that the median feature map encodes in its X- and Y-directions. Subsequently, along the spatial dimension, the output feature map is divided into the first two groups. Subsequently, each group undergoes a convolution operation to convert the tensors to have the same number of channels. Equations (14) and (15) are the result of recalculating the x and y coordinates of the raw map features following the application of the sigmoid operation.

$$g^h = \sigma\left(F_h\left(f^h\right)\right) \quad (14)$$

$$g^W = \sigma\left(F_W\left(f^W\right)\right) \quad (15)$$

Where σ is the sigmoid function, and f^h and f^W are each group's pre-transformation output feature maps. In equation (16), the final map of features y is defined.

$$y_c(i, j) = x_c(i, j) \times g_c^h(i) \times g_c^W(j) \quad (16)$$

where g^h and g^W these are the expanded output maps of features used as weights for the attention mechanism.

3.4.3. Adaptive Feature Aggregation Module

To exploit complementary features, the global- and local-level features of the non-local attention module and the Coordinate Attention module are adaptively fused. Global features can help local features identify important features by providing information about texture features and shape descriptors. Both local and global features complement each other: local features provide a wealth of spatial information, while global features lack significant spatial information. Consequently, the study proposes a squeeze-and-excitation (SE)-based module for adaptive feature aggregation to guide the fusion of features from neighbouring layers.

To obtain significant correlations between the channels, the feature maps of the neighbouring layers are first linked, then routed through the SE layer. After that, a pointwise convolution is applied to these feature maps to reduce the total number of filters. Channel-wise attention features are then globally extracted by applying a global average pooling layer. The softmax function receives the feature map, then suppresses unimportant background sounds and retains only the crucial data. Moreover, to enhance feature representation power and accurate localisation, the high-level features are appended to reweighted low-level features. Equation (17) is the formulation for this operation.

$$\eta^{(t)} = \eta_h^{(t+1)} \oplus \left(\eta_l^{(t)} \otimes \sigma\left(\mathrm{GAP}\left(F\left(\eta_f^{(t)}\right)\right)\right)\right) \quad (17)$$

where $\eta_f^{(t)} = \mathrm{SE}\left(\left[\eta_l^{(t)}, \eta_f^{(t+1)}\right]\right)$,$\oplus$ and $\otimes$ represent summation and multiplication of elements, and F stands for the 1x1 convolution layer. After the classification process is complete, tuning should be performed to achieve the best possible accuracy in detecting crime images. Therefore, in this work, LBOA has been used for accurate detection.

3.4.4. Hyperparameter Tuning using LBOA

The tuning process plays a crucial role in accurate crime detection. Hence, a novel optimisation called LBOA is introduced. This section describes the motivation behind the suggested LBOA and presents its mathematical modelling for use in tuning applications [27]; [28].

3.4.5. Inspiration of LBOA

The Superb Lyrebird, as well as Albert's Lyrebird, are the two species of Lyrebirds that are native to Australia. The Menuridae family includes this incredible bird [29]. Most people are aware of them for their remarkable ability to mimic sounds from their environment, both natural and artificial, and for the stunning beauty of a male bird's tail when it is spread during a mating ritual. The most famous native birds of Australia are Lyrebirds, which have distinctive plumes of neutral-coloured tail feathers. The length of a female Superb Lyrebird is 74-84cm, while a male typically measures 80-98cm. Albert's Lyrebird females can grow up to 84cm in length, while the maximum size of a male is 90cm. In the meantime, the female is slightly smaller. Superb Lyrebirds and Albert's Lyrebirds are similar except for the Less spectacular and smaller flying feathers of Albert's Lyrebird. Superb Lyrebirds weigh slightly more at 0.97 kg, compared to their approximate weight of 0.93kg. When the Lyrebird detects possible danger, one of its behavioural traits is visible. This bird pauses in this situation, carefully surveys its surroundings, and then seeks refuge elsewhere or leaves the area. The suggested LBOA technique, described below, was developed through mathematical modelling of this Lyrebird strategy in times of peril.

3.4.6. Algorithm Initialisation

Lyrebirds make up the population in the population-based metaheuristic algorithm known as the proposed LBOA approach. By leveraging its members' collective search power when tuning problems, the LBOA can provide suitable solutions within the problem-solving area. Each Lyrebird selects the choice variables it uses as an LBOA participant based on its position in the problem-solving hierarchy. Every Lyrebird can therefore be represented mathematically as a vector, with each component of the decision variable represented by the vector. Together, LBOA members comprise the algorithm's population, which can be represented mathematically by a matrix, as shown in Equation (18). Equation (19) is used to randomly initialise each LBOA member's position in the problem-solving space.

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_i \\ \vdots \\ X_N \end{bmatrix}_{N\times m} = \begin{bmatrix} x_{1,1} & \cdots & x_{1,d} & \cdots & x_{1,m} \\ \vdots & \ddots & \vdots & . & \vdots \\ x_{i,1} & \cdots & x_{i,d} & \cdots & x_{I\prime m} \\ \vdots & \cdots & \vdots & \ddots & \vdots \\ x_{N,1} & \cdots & x_{N,d} & \cdots & x_{N,m} \end{bmatrix}_{N\times m} \quad (18)$$

$$x_{i,d} = lb_d + r \cdot (ub_d - lb_d) \quad (19)$$

Here, X is the population matrix of LBOA, X_i is the i th LBOA member (candidate solution), $x_{i,d}$ is its d th dimension (decision variable) in the search space, N is the quantity of Lyrebirds, m is the quantity of variables used in a decision, r is an interval-based random number $(0,1)$, lb_d, and ub_d are, in turn, the decision variable's upper and lower bounds for the d th.. Assessing the problem's objective function is feasible, as every member of the LBOA represents a potential solution to the issue. Consequently, the objective function values are accessible and proportional to the total population. According to equation (20), a vector can represent the grouping of assessed values for the problem's objective function.

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_i \\ \vdots \\ F_N \end{bmatrix}_{N\times 1} = \begin{bmatrix} F(X_1) \\ \vdots \\ F(X_i) \\ \vdots \\ F(X_N) \end{bmatrix}_{N\times 1} \quad (20)$$

Here, F is the goal function's evaluated vector, and F_i the i-th member of the LBOA determines the assessed objective function. A useful metric for assessing the calibre of the potential solutions is the objective function's evaluated values. There exists a correlation between the best-examined value for the objective function and the best candidate solution, also known as the best LBOA member, and the worst-examined value for the objective function and the worst LBOA member. In addition, as the locations of the Lyrebirds in the problem-solving space change with each iteration, the best possible candidate solution should also change, as assessed by comparing objective function values.

3.4.7. Mathematical Modelling of LBOA

The suggested LBOA approach updates the positions of population members at every cycle, according to the Lyrebird strategy's mathematical modelling, whenever they detect danger. The population updating process consists of two stages: (i) fleeing and (ii) hiding, depending on the Lyrebird's choice in this scenario. Equation (21) simulates the Lyrebird's decision-making process in the LBOA design when it must select between hiding and escaping from danger. As a result, each iteration modifies only the position of each LBOA member in accordance with the first and second phases.

$$\text{Update process for } X_i: \begin{cases} \text{based on Phase 1}, r_p \le 0.5 \\ \text{based on Phase 2, else} \end{cases} \quad (21)$$

Here, r_p is an arbitrary number within the range (0,1).

3.4.8. Phase 1: Escaping Strategy (Exploration Phase)

At this phase of the LBOA process, every population member's position in the search space is updated by a model that replicates the Lyrebird's flight from the dangerous to the safe areas. The Lyrebird shifts positions significantly and explores the problem-solving space after being relocated to a secure location, demonstrating the LBOA's capacity for global search. A member's position within the population is considered a safe area in the LBOA design if it contains other members with higher objective function values. For each LBOA member, the set of safe areas can therefore be ascertained using equation (22).

$$SA_i = \{X_k,\ F_k < F_i \text{ and } k \in \{1,2,..,N\}\},\ \text{ where } i = 1,2,\dots,N \quad (22)$$

Here, SA_i is the group of locations where the Lyrebird and X_k is the X matrix's kth row, which has a higher value for the objective function (i.e., F_k) than the i th LOA member (i.e., $F_k < F_i$). According to the LBOA design, the Lyrebird is believed to flee to one of these secure locations willfully. Equation (23) determines a new location for every LOA member depending on the Lyrebird displacement modelling conducted during this phase. If the value of the objective function improves, equation (24) states that this new one replaces the corresponding member's prior position.

$$x_{i,j}^{P1} = x_{i,j} + r_{i,j} \cdot \left(SSA_{i,j} - I_{i,j} \cdot x_{i,j}\right) \quad (23)$$

$$X_i = \begin{cases} X_i^{P1}, & F_i^{P1} \le F_{i,} \\ X_i, & \text{else,} \end{cases} \quad (24)$$

Here, SSA_i is the designated Lyrebird safe area, $SSA_{i,j}$ is its jth dimension, X_i^{P1} is the new location for the Ith Lyrebird determined using the suggested LOA's escape plan, $x_{i,j}^{P1}$ is its jth dimension, F_i^{P1} is the value of its objective function, $r_{i,j}$ are arbitrary values drawn from the range (0,1), and $I_{i,j}$ are randomly chosen numbers, either 1 or 2.

3.4.9. Phase 2: Hiding Technique (Phase of Exploitation)

During this stage of LBOA, the population member's position is modified in the search space according to the Lyrebird's modelling strategy for hiding in its immediate safe area. The Lyrebird's position changes slightly as it moves in small steps to find a good hiding place while accurately scanning its surroundings, demonstrating the potential of LBOA for local search. Equation (25), in LBOA design, is used to model the Lyrebird's migration regarding the closest suitable hiding place, thereby determining a new position for each member of the LBOA. If the objective function's value rises in this new position as per equation (26), then it takes the place of the corresponding member's previous position.

$$x_{i,j}^{P2} = x_{i,j} + \left(1 - 2r_{i,j}\right) \cdot \frac{ub_j - lb_j}{t} \quad (25)$$

$$X_i = \begin{cases} X_i^{P2}, & F_i^{P2} \le F_i \\ X_i, & \text{else} \end{cases} \quad (26)$$

Here, X_i^{P2} is the new location determined for the Ith Lyrebird using the suggested LBOA's hiding technique, $x_{i,j}^{P2}$ is its jth dimension, F_i^{P2} is the value of its objective function, $r_{i,j}$ are arbitrary values within the range (0,1), and the iteration counter is denoted by t.

3.4.10. Repetition Process, Pseudocode, and Flowchart of LOA

The first iteration of LBOA is complete when all Lyrebird positions have been updated. After that, the algorithm proceeds to the next iteration, and the LBOA population is updated using equations (21)–(26) until the final iteration is reached. The optimal resolution is adjusted and preserved at the conclusion of every cycle. The optimal candidate was selected from the algorithm's output after the complete application of LBOA as the solution to the problem. Algorithm 1 presents the LBOA implementation steps as pseudocode. The following is the work process, based on the LBOA flowchart: first, the algorithm is fed problem information on choice variables, constraints, and the objective function. Next, the number of iterations required to solve the given problem and the population size are calculated. The algorithm's initial population is generated randomly in the first step, and it is then evaluated based on the problem's intended purpose. The algorithm's initial version is launched after the initialisation phase. Subsequently, the initial Lyrebird's position within the problem-solving area is updated.

The Lyrebird has two options when it comes to escaping danger, as outlined in LBOA modelling: (i) hiding and (ii) escaping. According to equation (21), each Lyrebird is assumed to randomly select one of these two strategies with equal probability in the LBOA design. If the Lyrebird decides to use the escape route, equations (22) – (25) determine where it is in the area for solving problems. Equations (25) and (26) ascertain the Lyrebird's course of action within the problem-solving area if it chooses the hiding approach. The location of the initial Lyrebird, or population member, has been successfully updated thus far. The locations of the other Lyrebirds are subsequently updated in the problem-solving area using the same steps as for the first Lyrebird. At this point, the algorithm's first iteration ends once every Lyrebird position in the area has been modified. Until this iteration is saved, the objective function's values are used to determine which candidate solution is best. The algorithm then advances to the next iteration. In the problem-solving domain, the procedure for updating Lyrebirds is essentially the same as in the previous iteration. This process is repeated up to the algorithm's last iteration. The output, which is the answer to the given issue, contains the best solution discovered across all algorithm iterations. This signifies that the algorithm's implementation is complete.

Algorithm 1: Pseudocode of LBOA

Start LBOA.
Input problem information: objective function, variables, and constraints.
Set LOA population size (N) and iterations (T).
Generate the first population matrix using equation (19) randomly. $x_{i,d} \leftarrow lb_d + r \cdot (ub_d - lb_d)$
Evaluate the objective function.
Determine the best candidate solution.
For $t = 1$ to T
For $i = 1$ to N
Determine the kind of defence mechanism used by Lyrebirds against predator attacks using equation (21).

$$X_i \leftarrow \begin{cases} \text{based on Phase 1}, r_p \le 0.5 \\ \text{based on Phase 2, else} \end{cases}$$

if $r_p \le \mathbf{0.5}$ (chose Phase 1)

Determine candidate safe areas for the ith Lyrebird based on equation (22).

$$SA_i \leftarrow \{X_k, F_k < F_i \text{ and } k \in 1,2, \dots, N\}$$

Calculate the new position of the i th LBOA member using equation (23).

$$x_{i,j}^{P1} \leftarrow x_{i,j} + r_{i,j} \cdot \left(SSA_{i,j} - I_{i,j} \cdot x_{i,j}\right)$$

Update the LBOA member using equation (24)

$$X_i \leftarrow \begin{cases} X_i^{P1}, & F_i^{P1} < F_i \\ X_i, & \text{else} \end{cases}$$

else (chose Phase 2)

Calculate the new position of the i th LBOA member using equation (25).

$$x_{i,j}^{P2} \leftarrow x_{i,j} + \left(1 - 2r_{i,j}\right) \cdot \frac{ub_j - lb_j}{t}$$

Update the LBOA member using equation (26).

$$X_i \leftarrow \begin{cases} X_i^{P2}, & F_i^{P2} < F_i \\ X_i, & \text{else} \end{cases}$$

end (if)

end (For $i = 1$ to N)

Save the best candidate solution so far.

end (For t = 1 to T)
Output the best quasi-optimal result that the LBOA could produce.
End LBOA

4. Results and Discussion

4.1. Experimental Setup

The simulation of the suggested model was carried out on a Windows 10 Pro computer. A computer equipped with an Intel(R) Core(TM) i5-6300U CPU at 2.40 GHz and 2.50 GHz, and 8 GB of RAM was used to measure computation time. An interactive, web-based, open-source IDE for scientific computing, called Jupiter Lab, was used in the experiment. An Anaconda environment was used to access JupyterLab. A Python and R distribution called Anaconda was created specifically for scientific computing, data science, and machine learning. It provides a convenient setting for handling dependencies, executing code, and managing packages.

4.2. Performance Metrics

ACC is a widely used metric for evaluating segmentation model performance. Equation (27), when applied to all samples, yields the percentage of correctly recognised samples.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \tag{27}$$

Equation (28), which presents the PR rate, assesses how well a model predicts positive samples among those it considers positive.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{28}$$

Equation (29), also known as the true positive rate (TPR), measures a prediction model's effectiveness at detecting actual positives. It is calculated as the proportion of real positive results to all true positive and negative outcomes.

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \tag{29}$$

Equation (30) defines the F1 score, which is the PR and RC weighted average that measures ACC. It provides an evaluation of the test's ability to distinguish between favourable and unfavourable outcomes.

$$\text{F1} = \frac{2\times\text{precision}\times\text{recall}}{\text{precision+recall}} \tag{30}$$

4.3. Classification Analysis

Table 1 shows performance metrics for various classes in a classification model. Each row represents a different class, such as “Abuse,” “Arrest,” “Arson,” “Assault,” and so on. Each metric provides insight into the model's effectiveness for particular classes. For instance, the “Abuse” class shows high values across most metrics, indicating that the model is effective at identifying abuse cases.

Table 1: Metrics analysis with various classes

Classes	Accuracy	Specificity	Precision	Recall	F1-Score
Abuse	99.12	98.65	97.89	99.32	97.48
Arrest	98.45	98.76	98.76	98.02	97.85
Arson	97.77	98.23	99.25	97.45	99.21
Assault	98.11	99.36	98.67	98.98	97.18
Road accidents	99.07	97.72	98.64	99.44	98.02
Burglary	97.16	97.95	97.14	97.01	98.75
Explosion	98.89	99.13	98.32	99.11	97.28
Fighting	99.35	98.41	98.54	99.43	98.68
Robbery	97.67	97.39	97.21	98.11	97.91

Shooting	98.03	98.78	98.86	98.56	97.74
Stealing	97.32	97.83	99.09	98.45	99.37
Shoplifting	99.02	99.22	98.14	98.10	97.31
Vandalism	98.94	97.17	97.55	99.28	98.89

It has exceptionally high recall (99.32%) and accuracy (99.12%). As an illustration of the model's ability to distinguish arson cases, the "Arson" class shows slightly lower precision (97.45%) and strong specificity (98.23%), with an F1-Score of 99.21%. Overall, this table highlights the strengths and potential of the classification model in detecting crime images, enabling a thorough evaluation of its efficacy across various classes. The performance of different classes is shown with metrics in Figure 3.

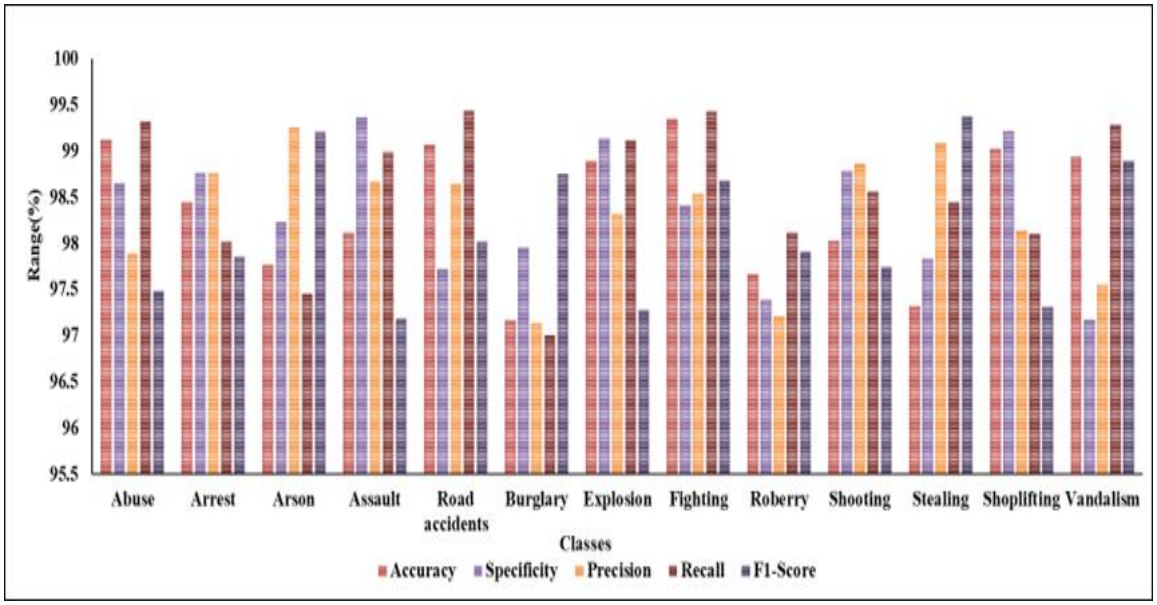

Figure 3: Metrics analysis across different classes

There are notable differences between the approaches, as indicated by the study's performance metrics in Table 2. The RNN achieved 90.81% accuracy, 90.85% specificity, 90.51% precision, 90.73% recall, and a 90.65% F1-score. The ANN performed better in the interim, achieving 91.57% accuracy, 91.68% specificity, 91.23% precision, 91.67% recall, and 91.35% F1-score. The Autoencoder method produced an F1-score of 92.33%, accuracy of 93.61%, specificity of 92.50%, precision of 92.28%, and recall of 92.58%, further improving these metrics.

Table 2: Training phase performance analysis on classification

Methods	Accuracy	Specificity	Precision	Recall	F1-Score
RNN	90.81	90.85	90.51	90.73	90.65
ANN	91.57	91.68	91.23	91.67	91.35
Autoencoder	93.61	92.50	92.28	92.58	92.33
ST	94.83	93.68	93.57	93.21	93.65
Proposed LBOA-based AdaHybridANet model	95.23	94.21	94.82	94.21	94.75

The ST method also outperformed earlier methods, achieving 94.83% accuracy, 93.68% specificity, 93.57% precision, 93.21% recall, and an F1-score of 93.65%. The AdaHybridANet model, based on the suggested LBOA, outperformed all others in terms of accuracy, specificity, precision, recall, and F1-score, achieving 95.23%, 94.21%, 94.82%, and 94.75%, respectively.

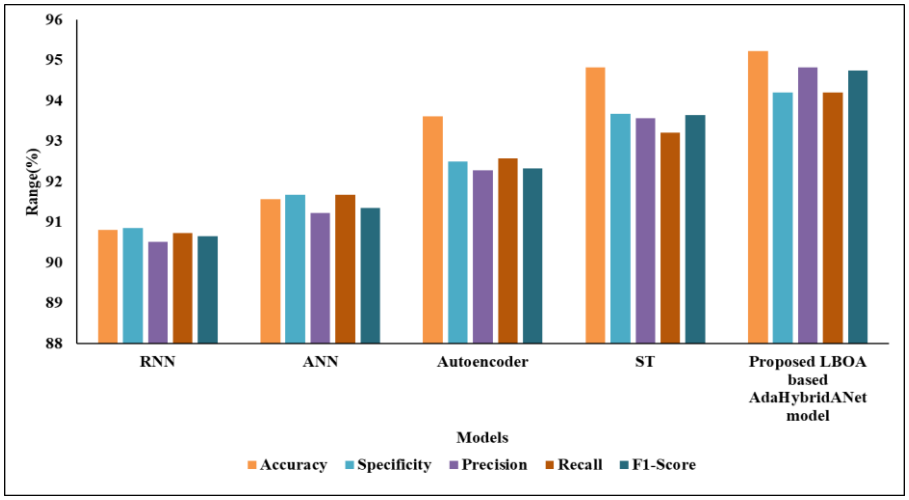

Figure 4: Performance analysis of the training phase in classification

This comprehensive analysis demonstrates how the proposed LBOA-based AdaHybridANet model outperforms other research methodologies. Figure 4 shows the classification performance during training for both the current and the suggested models.

Table 3: Testing phase performance analysis on classification

Methods	Acc	Spec	Pr	Rc	F1
RNN	95.36	94.87	94.32	93.84	94.63
ANN	96.11	92.34	94.09	94.85	97.35
Autoencoder	97.22	97.14	97.13	96.92	97.35
ST	97.85	97.75	97.56	97.96	96.68
Proposed LBOA-based Ada Hybrid a Net model	99.32	99.13	98.79	98.98	98.76

In Table 3, various performance measures are reported for different approaches. The results of the RNN approach are as follows: 94.63% F1-score, 93.84% recall (Rc), 94.32% precision (Pr), 94.87% specificity (Spec), and 95.36% accuracy (Acc).

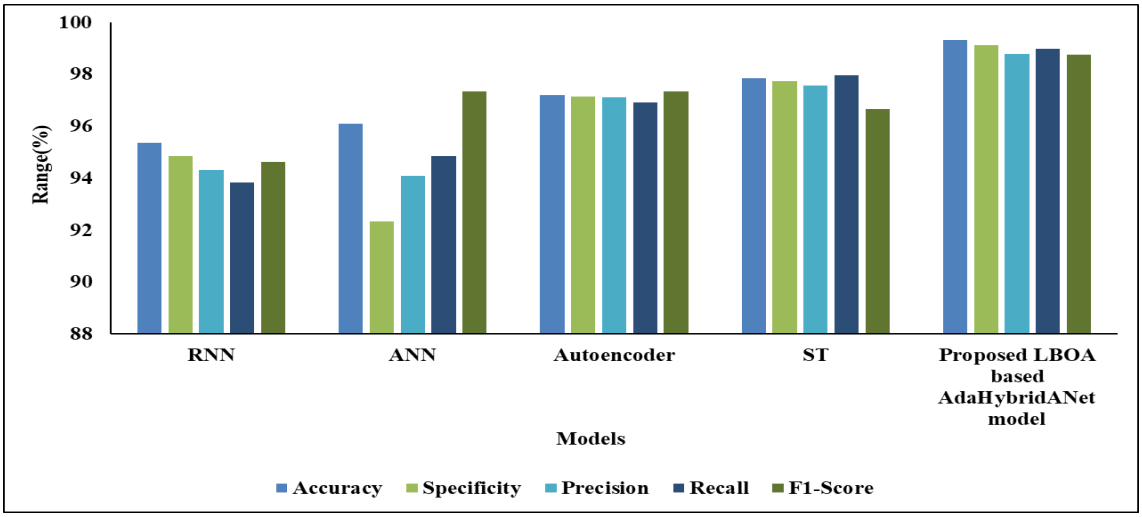

Figure 5: Training phase performance analysis on classification

As of now, the ANN method yields an F1-score of 97.35%, an accuracy of 96.11%, a specificity of 92.34%, a precision of 94.09%, and a recall of 94.85%. With 97.22% accuracy, 97.14% specificity, 97.13% precision, 96.92% recall, and a consistent F1-score of 97.35%, the Autoencoder method demonstrates significant improvements across these metrics. The ST method yields higher overall values as well, with 97.85% accuracy, 97.75% specificity, 97.56% precision, 96.76% recall, and 96.68% F1-score. The AdaHybridANet model, which is based on LBOA, exhibits superior performance compared to other methods.

Its F1-score is 98.76%, accuracy is 99.32%, specificity is 99.13%, precision is 98.79%, and recall is 98.98%. These findings demonstrate the superiority of the proposed LBOA-based AdaHybridANet model and highlight its remarkable efficacy across the assessed crime-detection metrics, compared with RNN, ANN, Autoencoder, and ST methods. Figure 5 illustrates classification performance during testing for both the current and the suggested models.

5. Conclusion

The crime images are taken from the UCF-Crime dataset. The FLLF preprocessing technique, which aims to enhance contrast and fine details in crime images, is introduced in the proposed study. ViTRes2DualNet is used in the extraction process. To increase the receptive field and improve crime detection, the feature data extracted from the two branches (ViT and Res2Net) was fused. Afterwards, AdaHybridANet is used in the classification process to identify important characteristics, and then adaptively combines them to leverage the complementary features of the attention modules. In contrast, the coordinate attention module focuses on extracting spatial features for precise localisation, while the ENLA module captures global features. The modules mentioned above enhance the network's capacity for feature representation and increase its generalizability. This paper presents the new bio-inspired metaheuristic algorithm, LBOA, which mimics the natural behaviour of Lyrebirds in the wild by adjusting the classifier's hyperparameters. The primary source of inspiration for LBOA is the way Lyrebirds choose to flee or hide in their environment by surveying their surroundings when they sense danger. The theory of LBOA was formulated and computationally represented in two stages: (i) investigation, which involved simulating the Lyrebird's escape plan, and (ii) utilisation, which involved simulating the Lyrebird's hiding plan. At 99.32% accuracy, the proposed LBOA-based AdaHybridANet outperforms current techniques. Further investigation into the potential applications of the proposed method is an intriguing avenue to pursue.

Acknowledgment: The authors sincerely extend their profound gratitude to New Horizon College of Engineering, Quest Technologies, Saranathan College of Engineering, and Ukrainian National Forestry University for their invaluable guidance, institutional support, and scholarly resources.

Data Availability Statement: All data generated or analyzed during this study are available from the corresponding author upon reasonable request. Access will be granted in accordance with ethical and privacy considerations relevant to the study.

Funding Statement: This research was conducted without any external financial support. No grants, sponsorships, or institutional funds were utilized in the development of this manuscript.

Conflicts of Interest Statement: The authors declare that there are no conflicts of interest—financial, academic, or personal—that could have influenced the outcomes or interpretation of this work.

Ethics and Consent Statement: This study conforms to established ethical standards and protocols. Informed consent was obtained from all participants prior to data collection.

References

1. L. Kirichenko, T. Radivilova, B. Sydorenko, and S. Yakovlev, “Detection of Shoplifting on Video Using a Hybrid Network,” *Computation*, vol. 10, no. 11, p. 199, 2022.
2. M. Q. Gandapur, “E2E-VSDL: End-to-end video surveillance-based deep learning model to detect and prevent criminal activities,” *Image and Vision Computing*, vol. 123, no. 7, p. 104467, 2022.
3. Z. Qin, H. Liu, B. Song, M. Alazab, and P. M. Kumar, “Detecting and preventing criminal activities in shopping malls using massive video surveillance based on deep learning models,” *Annals of Operations Research*, vol. 326, no. 7, pp. 9, 2021.
4. P. Wu, J. Liu, Y. Shi, Y. Sun, F. Shao, Z. Wu, and Z. Yang, “Not only look, but also listen: Learning multimodal violence detection under weak supervision,” *in Proc. European Conf. on Computer Vision (ECCV)*, Glasgow, United Kingdom, 2020.
5. W. Ullah, A. Ullah, I. U. Haq, K. Muhammad, M. Sajjad, and S. W. Baik, “CNN features with bi-directional LSTM for real-time anomaly detection in surveillance networks,” *Multimedia Tools and Applications*, vol. 80, no. 11, pp. 16979–16995, 2021.
6. W. Lin, H. Liu, S. Liu, Y. Li, R. Qian, T. Wang, N. Xu, H. Xiong, Q. Guo-Jun, and N. Sebe, “Human in events: A large-scale benchmark for human-centric video analysis in complex events,” *arXiv preprint, arXiv:2005.04490*, 2020. Available: https://arxiv.org/abs/2005.04490 [Accessed by 02/11/2024].

7. C. Chen, Y. Xie, S. Lin, A. Yao, G. Jiang, W. Zhang, Y. Qu, R. Qian, B. Ren, and L. Ma, "Comprehensive Regularization in a Bi-directional Predictive Network for Video Anomaly Detection," *in Proc. 36th AAAI Conf. on Artificial Intelligence*, Vancouver, Canada, 2022.
8. J. Yu, Y. Lee, K. C. Yow, M. Jeon, and W. Pedrycz, "Abnormal event detection and localization via adversarial event prediction," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 8, pp. 3572–3586, 2021.
9. X. Wang, Z. Che, K. Yang, B. Jiang, J. Tang, J. Ye, J. Wang, and Q. Qi "Robust unsupervised video anomaly detection by multipath frame prediction," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 6, pp. 2301–2312, 2021.
10. Z. Liu, Y. Nie, C. Long, Q. Zhang, and G. Li, "A hybrid video anomaly detection framework via memory-augmented flow reconstruction and flow-guided frame prediction," *in Proc. IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, Montreal, Quebec, Canada, 2021.
11. M. I. Georgescu, A. Barbalau, R. T. Ionescu, F. S. Khan, M. Popescu, and M. Shah, "Anomaly detection in video via self-supervised and multi-task learning*," in Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Nashville, Tennessee, United States of America, 2021.
12. R. Cai, H. Zhang, W. Liu, S. Gao, and Z. Hao, "Appearance-motion memory consistency network for video anomaly detection," *in Proc. AAAI Conf. on Artificial Intelligence*, Vancouver, Canada, 2021.
13. F. Landi, C. G. M. Snoek, and R. Cucchiara, "Anomaly locality in video surveillance," *arXiv preprint, arXiv:1901.10364*, 2019. Available: https://arxiv.org/pdf/1901.10364 [Accessed by 12//11/2024].
14. A. Nazir, R. Mitra, H. Sulieman, and F. Kamalov, "Suspicious behavior detection with temporal feature extraction and time-series classification for shoplifting crime prevention," *Sensors*, vol. 23, no. 13, p. 5811, 2023.
15. I. Muneer, M. S. Khatana, Z. Habib, and H. G. Mohamed, "Shoplifting detection using hybrid neural network CNN-BiLSMT and development of benchmark dataset," *Applied Sciences*, vol. 13, no. 14, p. 8341, 2023.
16. A. Mujahid, M. Aslam, M. U. G. Khan, A. M. Martinez-Enriquez, and N. U. Haq, "Multi-class confidence detection using deep learning approach," *Applied Sciences*, vol. 13, no. 9, p. 5567, 2023.
17. M. H. Lye, N. AlDahoul, and H. Abdul Karim, "Fusion of appearance and motion features for daily activity recognition from egocentric perspective," *Sensors*, vol. 23, no. 15, p. 6804, 2023.
18. S. Singla and R. Chadha, "Detecting criminal activities from CCTV by using object detection and machine learning algorithms," *in Proc. 2023 3rd Int. Conf. on Intelligent Technologies (CONIT)*, Hubli, India, 2023.
19. G. Gupta, P. Aggarwal, A. Jain, P. S. Lamba, A. K. Dubey, and G. Chaudhary, "Crime anomaly detection using CNN and ensemble model," *Fusion: Practice and Applications*, vol. 11, no. 1, pp. 89–99, 2023.
20. J. H. Jeong, H. H. Jung, Y. H. Choi, S. H. Park, and M. S. Kim, "Intelligent complementary multimodal fusion for anomaly surveillance and security system," *Sensors*, vol. 23, no. 22, p. 9214, 2023.
21. T. Yuan, X. Zhang, K. Liu, B. Liu, C. Chen, and J. Jin, "Towards Surveillance Video-and-Language Understanding: New Dataset, Baselines, and Challenges," *in 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, Washington, United States of America, 2024.
22. H. Ullah, Y. Zhao, F. Y. O. Abdalla, and L. Wu, "Fast local Laplacian filtering-based enhanced medical image fusion using parameter-adaptive PCNN and local features-based fuzzy weighted matrices," *Applied Intelligence*, vol. 52, no. 10, pp. 7965–7984, 2022.
23. P. Dutta, K. A. Sathi, M. A. Hossain, and M. A. A. Dewan, "Conv-ViT: A convolution and vision transformer-based hybrid feature extraction method for retinal disease detection," *Journal of Imaging*, vol. 9, no. 7, p. 140, 2023.
24. A. Tabbakh and S. S. Barpanda, "A deep features extraction model based on the transfer learning model and vision transformer 'TLMViT' for plant disease classification," *IEEE Access*, vol. 11, no. 5, pp. 45377 – 45392, 2023.
25. S. H. Gao, M. M. Cheng, K. Zhao, X. Y. Zhang, M. H. Yang, and P. Torr, "Res2Net: A new multi-scale backbone architecture," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 2, pp. 652–662, 2021.
26. I. Thayumanasamy and K. Ramamurthy, "Performance analysis of machine learning and deep learning models for classification of Alzheimer's disease from brain MRI," *Traitement du Signal*, vol. 39, no. 6, pp. 1961–1970, 2022.
27. D. Kilichev and W. Kim, "Hyperparameter optimization for 1D-CNN-based network intrusion detection using GA and PSO," *Mathematics*, vol. 11, no. 17, p. 3724, 2023.
28. D. Ujwal, M. S. Koti, and R. B. Sulaiman, "Machine learning approach for cyberbullying identification: A gradient boosting and Flask-based implementation," *AVE Trends in Intelligent Computer Letters*, vol. 1, no. 2, pp. 95–103, 2025.
29. A. Thirunagalingam, "Bias detection and mitigation in data pipelines: Ensuring fairness and accuracy in machine learning," *AVE Trends in Intelligent Computing Systems*, vol. 1, no. 2, pp. 116–127, 2024.
30. F. K. Karim, D. S. Khafaga, M. M. Eid, S. K. Towfek, and H. K. Alkahtani, "A novel bio-inspired optimization algorithm design for wind power engineering applications time-series forecasting," *Biomimetics*, vol. 8, no. 3, p. 321, 2023.
31. M. A. Sayedelahl and R. M. Farouk, "Hybrid approach to image segmentation with artificial neural networks and Gabor wavelets," *AVE Trends in Intelligent Computing Systems*, vol. 1, no. 2, pp. 77–90, 2024.