Journal of Theoretical and Applied Information Technology
30th April 2025. Vol.103. No.8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A NOVEL ENTROPY-BASED CASCADED CAPSULE NEURAL NETWORK WITH AN OPTIMIZED LSTM FOR ANOMALY SEGMENTATION AND CLASSIFICATION

[1*]SHAMEEM AKTHAR K, [2]DR. K. LAKSHMI PRIYA

[1*]Research Scholar, Department of Computer Science, Karpagam Academy of Higher Education, Coimbatore, Tamil Nadu-641021, India

[2]Associate Professor, Department of Computer Science, Karpagam Academy of Higher Education, Coimbatore, Tamil Nadu-64102, India

[1*]Corresponding Author email id: s.aktharu@gmail.com

ABSTRACT

The topic at hand of Anomaly Detection (AD) is important and well-researched. But creating successful AD techniques for complex and high-dimensional data is still difficult. There exist various methods for the detection of anomalies. Though, there are various existing methods, to improve the efficiency and to bring more advantages a method called Entropy based cascaded capsule neural network (CCNN) with an optimized LSTM (OLSTM) for segmentation and classification is suggested for the AD in the videos. Initially, pre-processing is done using the Gaussian filter followed by the segmentation using the Cascaded Capsule Neural network. Using this segmented data along with the dataset classified using entropy, Feature extraction (FE) is done by which the features are extracted. Finally, classification is done using the Optimized LSTM using ROCO algorithm method. The suggested approach has an accuracy of 98%, specificity of 99.8%, sensitivity of 92.9%, Precision of 92.9, F measure of 92.9%, FNR of 10%, FPR of 0.3%and Matthew Correlation Coefficient (MCC) of 89%. Thus, from the results, it is seen that our suggested approach performed more effectively than the other methods that are currently in use.

Keywords: *Anomaly Detection, Entropy, CCNN, OLSTM, ROCO*

1. INTRODUCTION

Data items that considerably differ from the majority of data objects are referred to as anomalies. Finding these abnormalities is the goal of AD, which has significant applicability in many different fields [1]. AD plays increasingly significant roles as a result of the growing demand and applications in numerous fields [2]. The type of procedure causing the anomaly should determine which AD algorithm is used. the methods for AD, especially in relation to cyber security. The main strategies can be divided into three categories: distance-based, density-based, and rank-based. The types of data that can be used with each of these methods include supervised, semi-supervised, and unsupervised data [3,4]. The following are some ways that AD is different from the conventional categorization issue: 1) It is exceedingly challenging to compile a list of every potential negative (anomaly) sample. 2) Due to the rarity, it is a difficult effort to gather enough negative samples. One of the most common techniques for achieving AD involves learning a model from films of normal events, then identifying the abnormal events that would deviate from the taught model [5]. At its most fundamental, AD is the detection of patterns that deviate from the system's expected norm. This straightforward interpretation is extremely difficult due to a variety of factors [6].

Particularly for AD, machine learning (ML) and deep learning (DL) are popular tools. Anomalies frequently result from intentional attack, sensor malfunction, and considerable environmental change, which the sensor registers as an abnormal state [7]. Surveillance is the method used to keep an eye on people's behavior and actions in order to manage and safeguard them. Using Internet of Things (IoT) gadgets like closed-circuit television (CCTV) cameras is the most common way to observe interesting objects from a distance. Artificial intelligence is implemented in IoT devices to improve quality of life [8]. Over the past few years, AD technique has drawn a lot of interest. It can be broadly divided into the following types: neighbor-based methods, dimensionality reduction-based methods, and statistical methods [9].

Video surveillance (VS) is the most common application of computer vision for observation and monitoring in both public and private settings. Intelligent VS systems are used to recognize, track, and analyze objects without human involvement. These intelligent VS systems find applications in homes, businesses, hospitals, malls, and parking lots, depending on the user's preferences. The majority of computer vision research focuses on topics such as scene comprehension and analysis, video analysis, anomaly/abnormality detection methods, human-object detection and tracking, activity recognition, facial expression recognition, urban traffic monitoring, human behavior monitoring, and detection of unusual events in surveillance scenes [10]. Detecting unusual events, such as traffic accidents, crimes, or illegal behavior, is a critical function in VS. Abnormal events occur significantly less frequently compared to normal activity. Therefore, it is crucial to develop intelligent computer vision algorithms for automatic video anomaly detection (AD) to save labor and time. The goal of a realistic AD system is to rapidly warn users when their behavior deviates from expected patterns and to specify the time window during which the anomaly occurs. AD can be thought of as a coarse degree of visual knowledge that distinguishes anomalies from typical patterns. [11]. With the aid of CCTV Cameras and related software, various agencies are using VS systems to uphold law and order and safeguard the safety of citizens and important assets [1]. However, using a human observer to spot every oddity in the video and analyze it would be impossible and unpractical. As a result, an intelligent VS system has been installed with several methods for localizing and detecting anomalies, which are closely related. While localization focuses on pinpointing the position of the anomaly using a bounding box, detection is the process of recognizing outliers within a frame [12]. A sophisticated device called the Surveillance Video Anomaly Detection (SVAD) system is intended to automatically identify suspicious or anomalous behavior in VS material. The technology works by examining the video frames and spotting variations from typical movement or activity patterns. The position of pixels in the video frame at the time of an event can be detected and analyzed using sophisticated algorithms and machine learning approaches, which are used to do this [13].

In recent years, the fundamental challenges associated with big data have garnered considerable attention. The five Vs of big data—value, veracity, variety, velocity, and volume include these. Value, veracity, and variety all allude to the advantages of data analysis. Veracity describes how accurate the data is, while variety describes the various sorts of data, such as structured, semi-structured, and unstructured data. Volume refers to the total amount of data being accumulated (i.e., the size of the data), whereas velocity denotes the "speed" at which the data are generated and may have many dimensions [14]. Every day, smart sensors and submeters installed in residential buildings produce massive amounts of data. If applied properly, this information could assist utility companies, energy suppliers, and end-users in recognizing anomalous patterns in power consumption and understanding their causes. As a result, anomaly detection (AD) could prevent a minor problem from worsening. Furthermore, it will facilitate better decision-making to reduce energy waste and promote ecologically and energy-efficient behavior [15]. The main contribution of the work as follows:

- The proposed approach uses Cascaded Capsule Neural Networks (CCNN) to improve anomaly detection by better capturing spatial hierarchies and preserving object relationships for more accurate localization compared to traditional CNN methods.
- Entropy-based feature selection enhances feature extraction by retaining only the most relevant features, reducing redundancy and computational complexity, and outperforming traditional methods in handling high-dimensional video data.
- The integration of Optimized LSTM (OLSTM) with the ROCO algorithm enhances anomaly classification by reducing overfitting, improving sequential dependency handling, and ensuring better real-time performance over conventional LSTM models.

2. RELATED WORKS

Chang, et. al., 2021 [16] have proposed a novel diagnostic technique for sensor data gathered throughout the production of semiconductors.

These signals offer crucial information for forecasting the final product's production and quality. Time series data for fault detection and classification (FDC) in real time make up a large portion of the data collected during this procedure. This implies that time series categorization (TSC) needs to be done when the product is being made. The ability to distinguish between normal and aberrant data has grown in importance as semiconductor production has advanced and new difficulties in identifying them have emerged.

Chan, et. al., 2021 [17] have created the "SegmentMeIfYouCan" benchmark, which is made up of fictitious data or has inconsistent labels. Their benchmark covers two tasks: road obstacle segmentation, which concentrates on any object on the road, whether it is known or unknown, and anomalous entity division, which takes into account any previously undiscovered object type. We offer two corresponding datasets and a test suite that conducts a thorough technique analysis, taking into account both new component-wise performance metrics that are indifferent to object sizes and well-established pixel-wise ones.

Liu, et. al., 2020 [18] have proposed a hybrid focused LSTM-CNN model (HALCM) and a two-level clustering-based QTS segmentation algorithm (TCQSA) as part of an autonomous QTS anomaly detection framework (AQADF). After the QTS is automatically divided into quasi-periods by TCQSA, HALCM classifies these periods as either normal or anomalous. TCQSA is notably noise-resistant and highly ubiquitous since it combines the k-means algorithm with hierarchical clustering. To simulate the fluctuation pattern of QTS, HALCM combines CNN and LSTM to concurrently extract local characteristics and general variation patterns.

Hansen, et. al., 2022 [19] have suggested a brand-new, anomaly detection-inspired method for few-shot medical picture segmentation that does not explicitly describe the background. Rather, anomaly scores are calculated for all query pixels using only one foreground prototype. A learnt threshold is then used to threshold these anomaly scores to perform the segmentation. Their suggested anomaly detection-inspired few-shot medical image segmentation technique uses super voxels to take use of the 3D structure of medical pictures, with the help of a novel self-supervision task.

Hassan, et. al., 2023 [20] have introduced a novel unsupervised anomaly instance segmentation approach that does not require any ground truth labels to identify baggage risks in X-ray scans as anomalies. Additionally, the framework only needs to be trained once because of its stylization capability, and it can identify and extract illegal items at the inferences stage regardless of the scanner's parameters. Using a suggested stylization loss function, their one-stage method first learns to recreate typical baggage content using an encoder–decoder network. The model then examines the differences among the initial and rebuilt images to determine the aberrant areas.

Dissanayake, et. al., 2020 [21] developed a strong classifier for identifying abnormal cardiac sounds. Additionally, acknowledging the urgent need for explainable AI models in the medical field, we also use model interpretation techniques to reveal hidden representations that the classifier has learnt. Based on experimental results, the model's ability to learn segmentation is crucial for classifying aberrant heart sounds.

Shi, et. al., 2021 [22] have suggested an efficient unsupervised anomaly segmentation method that is capable of identifying and separating abnormalities in constrained and tiny areas of images. For each subregion of an image, we create a multi-scale region feature generator that can produce numerous spatial context-aware representations using deep convolutional networks that have already been trained. The regional representations are discriminative and highly useful for anomaly detection since they encode the regional background information in addition to describing the local properties of the associated areas.

Although detection of anomalies and forecasting maintenance techniques can be used to create more intelligent and low-risk approaches for service scheduling, industrial implementations of these techniques are still rare because of the challenges of achieving satisfactory results in real-world situations. As a result, applications of these techniques in stamping processes are rarely found. Accordingly, Coelho et al., 2022 [23] combined two different methods: Time segmentation combined with anomaly detection and feature dimension reduction (a) and machine learning classification methods (b) for efficient downtime prediction.

Song, et. al., 2024 [24] have introduced a quick segmentation-based method for identifying

Journal of Theoretical and Applied Information Technology
30th April 2025. Vol.103. No.8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

surface flaws. The suggested model, which is based on a modified U-Net, applies a hybrid residual module (SAFM) to the decoder structure by substituting feedforward neural networks and an enhanced spatial attention mechanism for the remaining downsampling levels, except the encoder's first Downsampling layer. Additionally, dilation convolutions are included in the decoding to decrease the vanishing gradient issue of the model and to acquire additional geographic data on the characteristic flaws.

Wei, et. al., 2024 [25] have suggested employing a Neural Gas network to simulate the distribution of features of normal images. This network provides the freedom to modify the topological structure to detect anomalies in the data flow. They use multi-scale feature embedding taken from a CNN pre-trained on ImageNet to get a robust representation, which improves performance with fewer training samples. Additionally, they presented an approach that can update parameters gradually without requiring the storage of prior samples.

3. PROPOSED METHODOLOGY

An effective approach is required to handle the complexity and high dimensionality of video anomaly detection (AD). To achieve this, an Entropy-based Cascaded Capsule Neural Network (CCNN) with an optimized LSTM (OLSTM) is proposed. The methodology follows a structured process consisting of four key stages. First, preprocessing is performed using a Gaussian filter to remove noise while preserving essential structural details in video frames. Next, segmentation is carried out using a Cascaded Capsule Neural Network (CCNN), which enhances spatial feature representation and captures hierarchical relationships between objects more effectively than traditional CNNs. Following segmentation, feature extraction is conducted using an entropy-based filtering technique, ensuring that only the most significant features are selected for analysis. Finally, classification is performed using an optimized LSTM (OLSTM) with the ROCO algorithm, which improves sequence learning, mitigates overfitting, and enhances computational efficiency. This comprehensive methodology significantly boosts accuracy and robustness in video anomaly detection. The detailed research method protocol, outlining each applied step, has been incorporated into the Methodology Section of the paper. Figure 1 shows the block representation of the suggested approach.

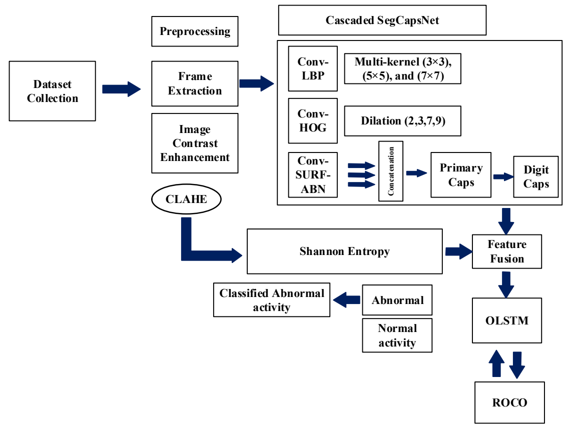

Figure1: Block illustration of suggested approach

3.1. Dataset Description

The UCF-Crime dataset consists of 128 hours of footage and is quite extensive. 1900 uncut, lengthy real-world surveillance footage are included, together with 13 actual anomalies such as vehicular crashes, burglaries, explosions, fights, robberies, and so on. The reason these anomalies were chosen is that they significantly affect the security of the society. This dataset can be used to accomplish two tasks. Firstly, there is general AD, which considers all abnormalities within one group and all routine activities within another. Secondly, it is utilized for recognizing each of the 13 peculiar actions.

3.2. Pre-processing

The initial step is the Pre-processing. It is a crucial component of numerous uses involving computer vision. Due to a number of elements, including image acquisition, illumination, poor contrast, and complex backgrounds, this stage attracted a lot of attention. The data pre-processing is a significant step for transforming the raw dataset into a most suitable format. The main benefit of an efficient preprocessing step is improved segmentation results, which lead to superior classification accuracy. The dataset may contain some redundant information and noise signal. Consider a dataset as $DS = DS\epsilon DS1, DS2, \dots DSn$. The pre-processing step of the system that is suggested employs a Gaussian filter to eliminate noise and smooth the pictures. The 2-D Gaussian distribution function employed by the proposed system's Gaussian filter for a pixel (i, j) is provided as,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$G(i,j) = DS\left(\frac{1}{2\pi\sigma^2}e^{\frac{-(i^2+j^2)}{2\sigma^2}}\right) = M \quad (1)$$

Standard deviation, $\sigma = \sqrt{\frac{\sum_{i=1}^{n}(DS-\mu)^2}{N}}$ (2)

Mean $\mu = \frac{\sum_{i=1}^{n} DS_i}{N}$ (3)

Where G→ Gaussian, M→Input (i/p) data, σ →standard deviation, N→ Total number of elements

3.3. Segmentation

Using the pre-processed data, Segmentation is performed. In our model, we have used a cascaded capsule neural network for the segmentation. It consists of i/p layer, convolution layer, Conv-LBP, Conv-HOG and Conv-SURF layer, capsule layer, Output (o/p) layer. To store the positions of items and their attributes within the image as well as for modelling their hierarchical relationships, capsule networks (CN) have been developed. A scalar value is the neural o/p of CNN. CN produces identically sized vectorial o/p but different routings. The parameters of the photographs are represented by a vector's routings.

3.3.1. Input layer

The initial layer in the network is i/player. The i/p information is given to the network by using this layer for which the segmentation is to be done.

3.3.2. Convolution layer

It consists of one or more convolution layers. This layer takes the features from the i/p data like edges, corners, textures and gradients.

3.3.2.1. Conv-LBP

Followed by the convolutional layers, add a Conv-LBP layer to compute Local Binary Patterns (LBP) for each pixel in the feature maps. It compares the data and encodes the results in a binary code. It produces a set of binary features that capture local texture patterns. A potent nonparametric operator for describing localised picture features is the LBP. An ordered binary set that is defined as LBP is obtained from a centre pixel $(x_c, y_c)$by comparing its grey value to the pixels of its eight neighbours. As a result, the LBP code is represented as an octet value in decimal form as,

$$z = LBP(x_c, y_c) = M\left(\sum_{n=0}^{7} S(i_n - i_c)2^n\right) \quad (4)$$

Where i_c →gray value of the center pixel (x_c, y_c), i_n→gray value of the pixels of its eight neighbors. t has been demonstrated that the Local Binary Pattern (LBP) code is invariant to all gray level transformations, ensuring that the transformed local neighborhood binary code remains unaltered.

$$S(i_n - i_c) = \begin{cases} 1 \; ; (i_n - i_c) \geq 0 \\ 0 \; ; (i_n - i_c) < 0 \end{cases} \quad (5)$$

3.3.2.2. Conv-HOG

This layer performs the gradient based features by analyzing local image gradients within small regions. It produces features capturing gradient orientation and magnitude information which are useful for shape and edge analysis. In the localized area of a picture, this method counts instances of gradient orientation. The HOG description emphasizes an object's structure or shape, outperforming other edge descriptors by computing features using both the magnitude and angle of the gradient. It generates histograms for the image's areas based on the magnitude and directions of the gradient. The image's gradient is computed. A grayscale image Im is required for determining the gradient. Each pixel's $G_x$and G_y are calculated, using,

$$G_x(f,g) = Im(f, g+1) - Im(f, g-1) \quad (6)$$

$$G_y(f,g) = Im(f-1, g) - Im(f+1, g) \quad (7)$$

Where f→ rows, g→ columns, Im →image

Once $G_x$and G_y is calculated, μ and θ of a pixel is using the equations shown below to determine the value.

$$\text{Magnitude } \mu = \sqrt{G_x^2 + G_y^2} \quad (8)$$

$$\text{Orientation, } \theta = tan^{-1}\frac{G_y}{G_x} \quad (9)$$

The window is divided into CxC -sized (C = 8) adjacent, exposed crawls. The HOGs for each trail are computed in orientations B (B = 9). A pixel whose direction is near to another orientation may be allocated another orientation because there are so few directions. To get over this issue, every single cell is given two close-crypts, and depending on how the pixel gradient is oriented from two near-directional directions, a little portion of the gradient's size μ decreases linearly. The histogram channels are evenly spread over 0-180^0 or 0-360^0 depending on whether the gradient is unsigned or signed. Each block having a size of $2C$x$2C$ pixels. When two blocks are covered by two routes in either a vertical or horizontal direction, the block has a step of C pixels. As a result, there are four blocks covering each cell. One characteristic value b is obtained in each block when the four-cell histograms are combined.

$$b = \frac{b}{\sqrt{\|b\|^2+\varepsilon}} \quad (9)$$

$\varepsilon \rightarrow$ minor positive constant to prevent division by zero in gradient-free units

To determine the HOG feature, all of the normalised blocks' features have been incorporated into a single vector.

$$h_g = M\left(\frac{h_g}{\sqrt{\left\|h_g\right\|^2+\varepsilon}}\ ;\min\left(h_n,\tau\right)\right) \quad (10)$$

For each cell, a histogram of edge
orientations are computed. The histogram channels are evenly
spread over 0–180□ć or 0–360□ć, depending on whether the
gradient is ‘unsigned’ or ‘signed’. The dimension number of
the histogram is 8 in one cell

For each cell, a histogram of edge
orientations are computed. The histogram channels are evenly
spread over 0–180□ć or 0–360□ć, depending on whether the
gradient is ‘unsigned’ or ‘signed’. The dimension number of
the histogram is 8 in one cell

For each cell, a histogram of edge
orientations is computed. The histogram channels are evenly
spread over 0–180□ć or 0–360□ć, depending on whether the
gradient is ‘unsigned’ or ‘signed’. The dimension number of
the histogram is 8 in one cell

For each cell, a histogram of edge
orientations are computed. The histogram channels are evenly
spread over 0–180□ć or 0–360□ć, depending on whether the
gradient is ‘unsigned’ or ‘signed’. The dimension number of
the histogram is 8 in one cell

For each cell, a histogram of edge
orientations is computed. The histogram channels are evenly
spread over 0–180□ć or 0–360□ć, depending on whether the
gradient is ‘unsigned’ or ‘signed’. The dimension number of
the histogram is 8 in one cell.

3.3.2.3. Conv-SURF

It performs the speeded up robust features and focuses on detecting interest points and local features. It is due to Hessian matrix H(x,σ) for higher Accuracy and chooses scales using the H value. Given a point $X = (f,g)$in an image I, $H(x,\sigma)$ at a level is stated as,

$$H(x,\sigma) = \begin{bmatrix} K_{ff}(x,\sigma) & K_{fg}(x,\sigma) \\ K_{fg}(x,\sigma) & K_{gg}(x,\sigma) \end{bmatrix} \quad (11)$$

Where$K_{ff}(x,\sigma)$ is the convolution of the Gaussian second order derivative $\frac{\partial^2}{\partial_x^2}g(\sigma)$ for I at X, and for $K_{fg}(x,\sigma)$ and $K_{gg}(x,\sigma)$. The expression for the Hessian's result is balanced using the relative weight w of the filter responses, which is typically set to 0.9. The maxima of this measure are used for SURF in order to identify the local spatial feature points.

$$d = \det\left(H_{approx}\right) = D_{xx}D_{yy} - \left(wD_{xy}\right)^2 M \quad (12)$$

$D_{xx} \rightarrow$ Approximations for the second-degree Gaussian partial derivative in the x-direction using weighted box filters. For D_{yy} and D_{xy}, measures can also be done.

The o/p of the cascaded Conv-LBP, Conv-HOG and Conv-SURF is given as,

$$y(m) = z + h + d \quad (13)$$

Finally, Conv-LBP, Conv-HOG, and Conv-SURF are cascaded and given to Capsule Neural Network (CCNN) to enhance the ability of the model

3.3.1.6. Primary caps (PC) and Digit caps (DC)

Squashing is a vectorial function of activation that is used in capsule neural networks. By using this squashing function as an activation function to make 10, 16D capsules, nonlinearity is achieved. The features from the convolution layers are o/p as scalars to this layer, sometimes known as the primary capsule layer, and all subsequent layers have to cope with 8D vector values.

$$v_j = \frac{\|s_j\|^2}{1+\|s_j\|}\frac{s_j}{\|s_j\|} \quad (14)$$

v_j→ o/p of the capsule j, S_j→ I/p of the capsule.

If an object is present in the image, v_j reduces long vectors in the direction of 1 and chokes short vectors in the direction of 0. The weighted sum of the prediction vectors $U_{j|i}$ in the capsules identified in the lower layers, with the exception of the first layer of capsule networks, is used to determine the total i/value of capsule S_j . In order to determine the prediction vector $U_{j|i}$, a lower-layer capsule's o/p O_i and a weight matrix W_{ij} are multiplied.

$$S_j = \sum_i b_{ij}\, U_{j|i} \quad (15)$$

$$U_{j|i} = W_{ij} O_i \quad (16)$$

b_{ij} → the dynamic routing procedure's coefficient and it is given as,

$$b_{ij} = \frac{exp(a_{ij})}{\sum_k exp(a_{ik})} y(m) \quad (17)$$

a_{ij}→log probability

The next layer to the PC layer is Digit caps layer has a completely connected layer performing the operation with i/p from each layer below it.

3.4. Feature Extraction

For many different study fields, feature extraction approaches have received a lot of interest. The most accurate and closely related features consistently generated the best categorization outcomes. Due to the fact that redundant and irrelevant features reduce the system's ability to accurately classify data, efficient and robust features are crucial in this step. Before feature extraction, there occurs feature fusion. It makes an effort to weed out superfluous information and extract the most discriminative information from a variety of i/p features. The segmented o/p and the i/p dataset's entropy-based image contrast enhancement is used to combine the features.

3.4.1. Image contrast enhancement (ICE)

This technique has been applied to improve visual appeal. Our model utilizes a method called Contrast Limited Adaptive Histogram Equalization (CLAHE). To address the issue of contrast over-amplification, CLAHE adapts Adaptive Histogram Equalization (AHE). CLAHE operates on distinct portions of the image, known as tiles, rather than analyzing the entire picture. Bilinear interpolation is then applied to combine adjacent tiles and eliminate artificial borders, enhancing contrast.

Essentially, CLAHE functions by limiting the contrast enhancement that is typically performed by conventional HE. Consequently, the functions controlling contrast enhancement also regulate the histogram's height and slope. Users can provide a clip limit (cl) or set the histogram's height to achieve desired results. We must first compute the average number of pixels before computing the histogram.

$$N_A = M\left(\frac{(N_x \times N_y)}{N_G}\right) \quad (18)$$

Where N_A→Average number of pixels, N_x→Total pixels in X, N_y→Total pixels in Y, N_G→ total gray levels

To clip the histogram cl must be calculated. It is given as,

$$N_{CL} = N_A \times N_{NCL} \quad (19)$$

N_{CL}→clip limit and N_{NCL}→ Normalized clip limit between 0 and 1. Afterwards, for each tile, the clip limit is applied for the height of histogram.

$$H_i = \begin{cases} N_{CL};\, if\, N_i \geq N_{CL} \\ N_i; \quad else \end{cases}; i = 1,2 \ldots L-1 \quad (20)$$

H_i → height of the histogram of i^{th} tile, N_i → histogram of the i^{th} tile and L→ total gray levels.

The total number of clipped pixels can be computed using the following formula,

$$N_c = (N_x \times N_y) - \sum_{i=0}^{L-1} H_i \quad (21)$$

N_c→ Number of clipped pixels

To compute the total pixels to be redistributed, the following formula is used,

$$N_R = \frac{N_c}{L} \quad (22)$$

The clipped histogram is normalized using,

$$H_i = \begin{cases} N_{CL};\, if\ N_i + N_R \geq N_{CL} \\ N_i + N_R; \quad else \end{cases} \; i = 1,2 \ldots L-1 \quad (23)$$

Until all the pixels are redistributed, the above step is repeated. The cumulative histogram is given as,

$$C_i = \frac{1}{(N_x \times N_y)} \sum_{j=0}^{i} H_j \quad (24)$$

The Shannon entropy can measure the uncertainty of a random process within a given probability distribution.

$$H(M) = -\sum_{i=1}^{N} p_i\, ln p_i \quad (25)$$

To provide the contextual region's histogram with a preset brightness and visual quality, uniform or exponential probability distributions are matched with it. Let P (f, g) be a pixel with

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

a value of s. Let R1, R2, R3, and R4 be the center points of four adjacent tiles. These four contextual zones are combined to provide a weighted total, and the tiles are blended for the output image. Artifacts between independent tiles are eliminated through bilinear interpolation. The new value of o, denoted as o', may be obtained using,

$$o' = (1-g)\big((1-f)\times R_1(o) + f\times R_2(o)\big) + g\big((1-f)\times R_3(o) + f\times R_4(o)\big)$$

$$(1-g)\big((1-f)\times R_1(o) + f\times R_2(o)\big) + g\big((1-f)\times R_3(o) + f\times R_4(o)\big) * H(M) \quad (26)$$

$$\mathrm{y(n)} = o' * H(M) \quad (27)$$

Thus, using this the enhanced images will be obtained.

3.4.2. Feature Extraction using Adaptive Batch normalization

The batch normalization (BN) is used to speed up the training. BN simply changes the mean and standard deviation of all the pixels in the feature maps in a convolution layer. Typically, it begins by z-score normalizing all pixels, multiplies the results by an arbitrary parameter alpha (scale), and then adds another arbitrary parameter beta (offset) before returning to the original values. Batch normalization is formulated as:

$$m` = \left(\frac{(m-\mu)}{\sigma} * \alpha\right) + \beta \quad (28)$$

The adaptive batch normalization is an extension of the batch normalization method based on the μ and σ of the mini batches individually, while BN is based on the entire mini batch.

The feature extraction can be done using the segmented o/p and the ICE-Entropy o/p. Thus, the o/p is given as,

$$m` = \left(\frac{(m-\mu)}{\sigma} * \alpha\right) + \beta * \big(y(m) + y(n)\big) \quad (29)$$

$y(m)$→segmented o/p, $y(n)$→ ICE-Entropy o/p

3.4. Classification

This is the final which is very important because it classifies and produces the final result based on the above process. It accepts the given i/pand produces the o/p (anomaly detected in videos or not). OLSTM-ROCO (Rider Optimizer Based Coati Optimization Algorithm) approach is used for the classification.

3.4.1. OLSTM-ROCO Algorithm

The forget gate, input gate, and output gate constitute the three gated units that make up the majority of the LSTM neural network. To preserve long-distance time series information dependence and accomplished high-precision prediction, these specialized gated units learn and retain sequence data. The forget gate regulates how well the current cell recalls previous information, while the input gate serves as the principal processor for incoming data. The output gate represents the output of the neuron. Assuming that the i/p sequence is $x_1, x_{2,..,} x_t$, the calculation formula for each LSTM neuron parameter at time t is as follows:

$$i_t = S(W_i * [h_{t-1}, x_t] + b_i) \quad (30)$$

$$f_t = S\big(W_f * [h_{t-1}, x_t] + b_f\big) \quad (31)$$

$$o_t = S(W_o * [c_t, h_{t-1}, x_t] + b_o) \quad (32)$$

$$c_t = f_t * C_{t-1} + i_t * tanh(W_c * [h_{t-1}, x_t]) \quad (33)$$

$$h_t = o_t * tanh(c_t) \quad (34)$$

Where i_t,f_t,o_t→i/p of the i/p gate, forget gate, o/p gate at time t respectively, x_t→i/p to the LSTM neuron at time t, h_{t-1}→o/p state of the hidden layer at time t-1, W_i, W_f, W_o →weight of the i/p and the i/p gate, the forget gate and the o/p gate of the neuron at time t and b_i, b_f, b_o→offset vectors, W_c →weight between the i/p and the cell unit, h_t→o/p of the hidden layer at time t and S→ sigmoid function.

The LSTM neural network's weights between nodes are optimized using a predetermined procedure, which can make the weights between neurons more rational and enhance the model's capacity for generalization and prediction.

The Coati Optimization Algorithm is characterized as a population-based metaheuristic, and coatis constitute members of this population. Therefore, the coatis' viewpoint provides a practical solution to address the challenge of COA. The coatis' initial position in the search space is randomly selected when the COA is applied [23].

$$\mathrm{X_i: x_{z,a} = lb_a + r.(u_{ba} - l_{ba})\,; z = 1,2,..N\ and\ a = 1,2,..m} \quad (35)$$

Where $X_i \rightarrow$position of the i^{th} coati in search space, $x_{z,a} \rightarrow$ value of a^{th} decision variable, $N \rightarrow$ number of coatis, $m \rightarrow$number of decision variables, $r \rightarrow$ random real number in the interval [0,1], $u_{ba} \rightarrow$ upper bound of the a^{th} decision variable, $l_{ba} \rightarrow$ lower bound of the a^{th} decision variable.

Modelling two of the coatis' natural behaviors forms the basis of the process of revising the location of coatis (candidate solutions) in the COA. It is stated that:

- Coatis use to attack iguanas
- Coatis' defense mechanisms against predators

As a result, the COA population is updated twice.

Phase 1: Iguana hunting and attacking strategy (exploration phase)

The first step of updating the coatis' population in the search area is illustrated by simulating the coatis' approach in attacking iguanas. Many coatis use this method to scale the tree in an attempt to frighten an iguana. More coatis gather around the fallen iguana, waiting beneath a tree. The coatis assault and pursue the iguana as soon as it lands. By enabling coatis to move to different locations inside the search region, this method illustrates the COA's global search capabilities within the problem-solving domain. In the COA design, the iguana is considered to be the best member of the population.

Additionally, it is thought that half of the coatis climb the tree while...As a result, the mathematical position of the coatis emerging from the tree is as follows:

$$X_i^{P1} = x_{z,a}^{P1} = x_{z,a} + r.\left(IG_a - I.x_{z,a}\right); for z = 1,2..,\frac{N}{2} \; and \; z = 1,2,..,m \tag{36}$$

The iguana is dropped to the ground and then placed randomly within the search area. The search space is defined, and coatis on the ground move in accordance with this random position.

$$IG^G: IG_j^G = lb_a + r.(u_{ba} - l_{ba}); \; a = 1,2,..m \tag{37}$$

$$X_i^{P1}: x_{z,a}^{P1} = \begin{cases} \left(x_{z,a} + r.\left(IG_a^G - I.x_{z,a}\right)\right); F_{IG^G} < F_i \\ \left(x_{z,a} + r.\left(x_{z,a} - IG_a^G\right)\right); \quad else \end{cases} \tag{38}$$

Where $z = \left[\frac{N}{2}\right] + 1, \left[\frac{N}{2}\right] + 2, ..N$ and $a = 1,2 \dots, m$

The formula, for this update condition for z=1, 2, N, can be used.

$$X_i = \begin{cases} X_i^{P1}; \; F_i^{P1} < F_i \\ X_i \quad \quad else \end{cases} \tag{39}$$

Where $X_i^{P1} \rightarrow$new position calculated for the i^{th} coati, $x_{z,a}^{P1} \rightarrow$in its a^{th} dimension, $F_i^{P1} \rightarrow$ objective function value, IG→ iguana's position in the search space, which actually refers to the position of the best member, $IG_a \rightarrow$ is in a^{th} dimension, $I \rightarrow$ Integer which is selected randomly, $IG^G \rightarrow$ position of the iguana on the ground which is randomly generated, $IG_a^G \rightarrow$in its a^{th} dimension, $F_{IG^G} \rightarrow$value of the objective function

(2) Phase 2: Process of escaping from predators (exploitation phase)

A statistical model is employed in the second step of updating the coatis' position in the search space. This model is based on the typical behavior of the species when facing and avoiding predators. When a coati is attacked by a predator, it flees its habitat. The COA's implementation of this technique has resulted in it being in a secure area close to its current location, showcasing the COA's capacity to leverage local search.

Based on Eqs. (40, 41), a random position is chosen close to where each coati is positioned in order to imitate this behaviour.

$$lb_a^{local} = \frac{lb_a}{t}, ub_a^{local} = \frac{ub_a}{t}\; ; t = 1,2,..T \tag{40}$$

Where t→ iteration counter

$$X_i^{P2}: x_{z,a}^{P2} = x_{z,a} + (1 - 2r).\left(lb_a^{local} + r.\left(ub_a^{local} - lb_a^{local}\right)\right) \tag{41}$$

$$where \; z = 1,2 \dots N \; and \; a = 1,2,..,m$$

If the newly computed position increases the value of the objective function, it is deemed acceptable, and the calculation is done using,

$$X_i = \begin{cases} X_i^{P2}; F_i^{P2} < F_i \\ X_i; \quad else \end{cases} \tag{42}$$

Four riders—bypass riders (B), followers (F), overtake riders (O), and attackers (A)—are used to characterize ROA. Each cyclist makes a strategic move in the direction of the destination [24].

1) B with the intention of avoiding them in order to reach the destination.

2) The F shall be obeyed and will be followed by the trailing rider.

3) The O keeps the leading rider in mind as it travels along its path to the destination.

4) The A advance to the target area while adhering to the principle of maximal speed.

Similarly, by using this algorithm, we can calculate the Speed of the coati and the success rate. It is calculated using the following equations,

The maximum speed, an i^{th} rider can drive will be calculated as,

$$X^i_{max} = \frac{M^i_A - M^i_I}{T_{off}} \tag{43}$$

M^i_A → maximum value of the i^{th} coati, and M^i_I→ minimum value of i^{th} coati, and $Toff$ →maximum time, which is allotted to reach the target position.

The success rate is calculated as,

$$S_r = \frac{1}{\|X_i - P_t\|} \tag{44}$$

Where X_i →i th coati position, P_t→ the target position.

Algorithm 1: ROCO algorithm for classification

Input=Information of optimization o/p=best solution for the input

Set parameters N and T (set i=t=1)
Create the initial population and calculate objective function
Update position of iguanas
If (i>N/2)
Generate position of the iguanas
Else
Calculate $X_i^{P1}$and update X_i
End if
If (i<N)
Repeat the steps
Else
Set i=1
Calculate $X_i^{P2}$and update X_i
End if
Save the best solution
Calculate X^i_{max}, S_r
Give the o/p of the best solution

4. RESULTS AND DISCUSSIONS:

In this section, using the selected dataset, the results are evaluated based on various performance metrics, including Accuracy, Sensitivity, Specificity, Precision, F-measure, False Negative Rate (FNR), False Positive Rate (FPR), and Matthews Correlation Coefficient (MCC). To validate the effectiveness of the proposed approach, a comparative analysis is conducted against existing techniques such as ROA [26], COA [27], Bi-LSTM [28], and ANN [29].

4.1. Dataset Description

The UCF-Crime dataset consists of 128 hours of footage and is quite extensive. 1900 uncut, lengthy real-world surveillance footage are included, together with 13 actual anomalies such as vehicular crashes, burglaries, explosions, fights, robberies, and so on. The reason these anomalies were chosen is that they significantly affect the security of the society. This dataset can be used to accomplish two tasks. Firstly, there is general AD, which considers all abnormalities within one group and all routine activities within another. Secondly, it is utilized for recognizing each of the 13 peculiar actions.

4.2. Performance Evaluation:

The proposed Entropy-Based Cascaded Capsule Neural Network (CCNN) with an optimized LSTM (OLSTM) demonstrates significant improvements over conventional methods such as ROA, COA, Bi-LSTM, and ANN, which are done using the performance metrics called Accuracy, Sensitivity, Specificity, Precision, F measure, FNR, FPR, and MCC.

Table 1 Evaluation of performance for 70% of training

Metrices	ROA	COA	Bi-LSTM	ANN	Proposed
Accuracy	0.9140	0.8189	0.9250	0.9165	0.9568
Precision	0.7510	0.5608	0.7729	0.7560	0.8812
Sensitivity	0.7510	0.5608	0.7729	0.7560	0.8812
Specificity	0.9684	0.9049	0.9757	0.9700	0.9820
F-Measure	0.7510	0.5608	0.7729	0.7560	0.8812
MCC	0.6423	0.3887	0.6716	0.6489	0.8308
NPV	0.9684	0.9049	0.9757	0.9700	0.9820
FPR	0.1086	0.1720	0.1013	0.1070	0.0503
FNR	0.3260	0.5162	0.3041	0.3210	0.1511

Table 1 presents the performance evaluation of different models (ROA, COA, Bi-LSTM, ANN, and the Proposed model) using 70% of the training data. The results show that the proposed model outperforms all other methods across all metrics. It achieves the highest accuracy (95.68%), precision, sensitivity, F-measure (all 88.12%), and specificity (98.20%), indicating strong predictive performance and low misclassification. Furthermore, it records the highest Matthews Correlation Coefficient (MCC) of 0.8308, showing robust correlation between predicted and actual classes. The proposed model also demonstrates the lowest False Positive Rate (FPR) and False Negative Rate (FNR), confirming its effectiveness and reliability compared to the other models evaluated.

Table 2 Evaluation Of Performance For 80% Of Training

Metrices	ROA	COA	Bi-LSTM	ANN	Proposed
Accuracy	0.9798	0.9579	0.9437	0.9363	0.980915
Precision	0.8826	0.8387	0.8104	0.7956	0.929395
Sensitivity	0.8826	0.8387	0.8104	0.7956	0.929395
Specificity	1.0122	0.9976	0.9882	0.9832	0.998088
F-Measure	0.8826	0.8387	0.8104	0.7956	0.929395
MCC	0.8177	0.7593	0.7215	0.7017	0.895048
NPV	1.0122	0.9976	0.9882	0.9832	0.998088
FPR	0.0648	0.0794	0.0888	0.0938	0.034347
FNR	0.1944	0.2383	0.2666	0.2814	0.1[illegible]

Table 2 shows the performance evaluation of various models (ROA, COA, Bi-LSTM, ANN, and the Proposed model) using 80% of the training data. The proposed model again demonstrates superior performance across all metrics. It achieves the highest accuracy (98.09%), precision, sensitivity, F-measure (all 92.94%), and specificity (99.81%), reflecting its strong capability to correctly identify both positive and negative cases. The model also achieves the highest MCC value (0.8950), indicating excellent predictive reliability. Additionally, it records the lowest False Positive Rate (FPR) and False Negative Rate (FNR), further confirming its robustness and efficiency over the other methods.

4.2.1. Accuracy:

It is the proportion of true forecasts to all i/p Observations. It is calculated using the following formula,

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ samples} \quad (45)$$

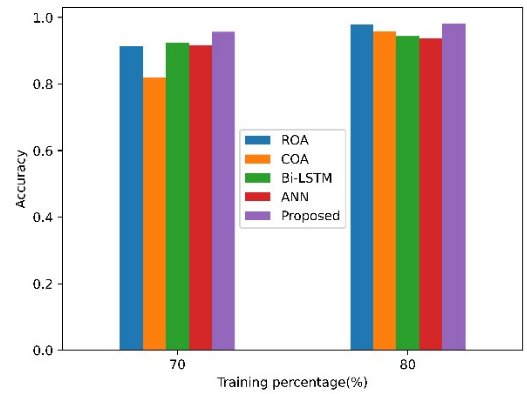

Figure 2: Evaluation Of Recommended And Existing Methods With Regard To Accuracy

Figure 2 provides the performance of the recommended and approaches in use with regard to Accuracy. From graph, the suggested approach has an accuracy of about 98% whereas the existing methods ROA, COA, Bi-LSTM and ANN has the accuracy of 97%, 95%, 94% and 93% respectively. Thus, from the graphical representation, this is evident that the proposed approach has a higher Accuracy.

4.2.2. Sensitivity

The fraction of real positives that are correctly identified is measured by sensitivity. It is given as,

$$Sensitivity = \frac{TP}{TP+FN} \quad (46)$$

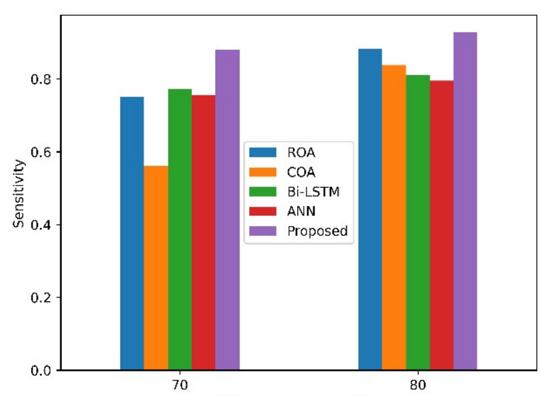

Figure 3: Evaluation Of Recommended And Existing Methods With Regard To Sensitivity

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 3 provides the performance of the recommended and approaches in use with regard to Sensitivity. From the graph, the proposed approach has a Sensitivity of about 92.9% whereas the existing methods ROA, COA, Bi-LSTM and ANN has the Sensitivity of 88%, 83%, 81% and 79% respectively. Thus, from the graphical representation, this is evident that the proposed approach has a higher Sensitivity.

4.2.3. Specificity

The percentage of real negatives that are accurately identified is measured by specificity. It is calculated using

$$\boldsymbol{Specificity} = \frac{\boldsymbol{TN}}{\boldsymbol{TN+FP}} \qquad (47)$$

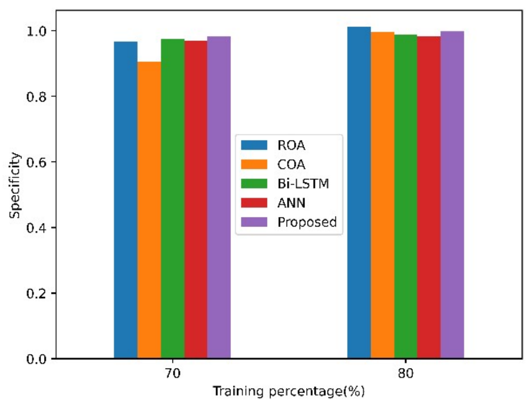

Figure 4: Evaluation Of Recommended And Existing Methods With Regard To Specificity

Figure 4 provides the performance of the recommended and approaches in use with regard to Specificity. From the graph, the proposed approach has a Specificity of about 99.8% whereas the existing methods ROA, COA, Bi-LSTM and ANN has the Specificity of 99%, 99%, 98% and 98% respectively. Thus, from the graphical representation, this is evident that the proposed approach has a higher Specificity.

4.2.4. Precision

How much of a model's positive predictions are actually right is determined by its precision, which is a performance indicator. In order to assess how well what you detect is actually present, precision is important. It is given as,

$$\boldsymbol{Precision} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}} \qquad (48)$$

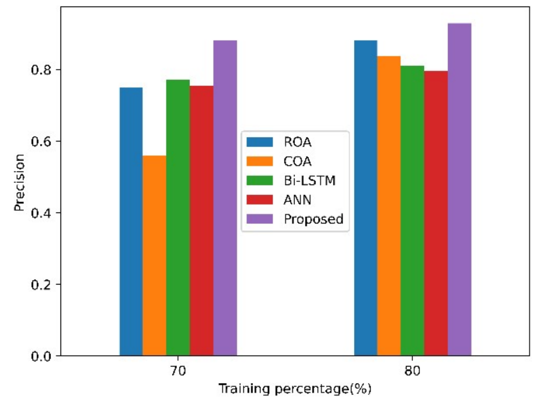

Figure 5: *Evaluation Of Recommended And Existing Methods With Regard To Precision*

Figure 5 provides the performance of the recommended and approaches in use with regard to Specificity. From the graph, the proposed approach has a Precision of about 92% whereas the existing methods ROA, COA, Bi-LSTM and ANN has the Precision of 88%, 83%, 81% and 79% respectively. Thus, from the graphical representation, this is evident that the suggested approach has a higher Precision.

4.2.5. F measure

A general score for performance evaluation, the F1-score is a combination statistic that combines Precision and recall. It is given as,

$$\textbf{F measure} = 2 * \frac{\boldsymbol{Precision*Recall}}{\boldsymbol{Precision+Recall}} \qquad (49)$$

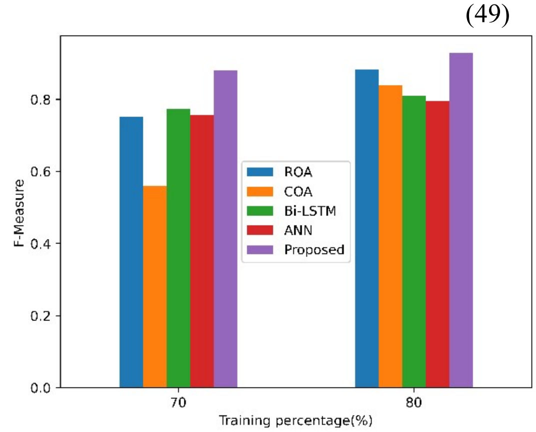

Figure 6: Evaluation Of Recommended And Existing Methods With Regard To F Measure

Figure 6 provides the performance of the recommended and approaches in use with regard to F measure. From the graph, the proposed approach has a F measure of about 92% whereas the existing methods ROA, COA, Bi-LSTM and ANN has the F measure of 88%, 83%, 81% and 79% respectively. Thus, from the graphical representation, this is evident that the suggested approach has a higher F measure.

4.2.6. False Negative Rate (FNR)

It refers to the values that are actually positive but predicted to negative. It is calculated using the formula,

$$\boldsymbol{FNR} = \frac{\boldsymbol{FN}}{\boldsymbol{FN+TP}} \quad (50)$$

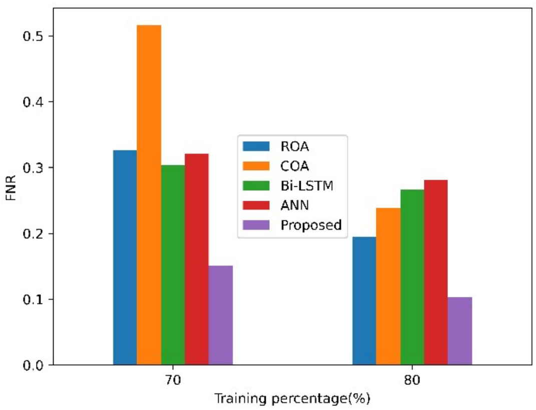

Figure 7: Evaluation of recommended and existing methods with regard to FNR

Figure 7 provides the performance of the recommended and approaches in use with regard to FNR. From the graph, the proposed approach has a FNR of about 10% whereas the existing methods ROA, COA, Bi-LSTM and ANN has the FNR of 19%, 23%, 26% and 28% respectively. Thus, from the graphical representation, this is evident that the proposed approach has a lower FNR.

4.2.7. False Positive Rate (FPR)

It refers to the values that are actually negative but predicted to be positive. It is calculated using the formula,

$$\boldsymbol{FPR} = \frac{\boldsymbol{FP}}{\boldsymbol{FP+TN}} \quad (51)$$

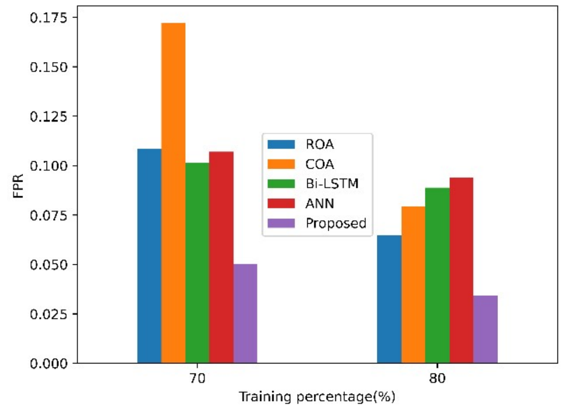

Figure 8: Evaluation of recommended and existing methods with regard to FPR

Figure 8 provides the performance of the recommended and approaches in use with regard to FPR. From the graph, the proposed approach has a FPR of about 0.3% whereas the existing methods ROA, COA, Bi-LSTM and ANN has the FPR of 0.6%, 0.7%, 0.8% and 0.9% respectively. Thus, from the graphical representation, this is evident that the proposed approach has a lower FPR.

4.2.8. Matthew Correlation Coefficient (MCC)

MCC measures the degree of correlation between expected and actual values. It's stated as,

$$\boldsymbol{MCC} = \frac{(\boldsymbol{TP}\times\boldsymbol{TN})-(\boldsymbol{FP}\times\boldsymbol{FN})}{\sqrt{(\boldsymbol{TP+FP})(\boldsymbol{TP+FN})(\boldsymbol{TN+FP})(\boldsymbol{TN+FN})}} \quad (52)$$

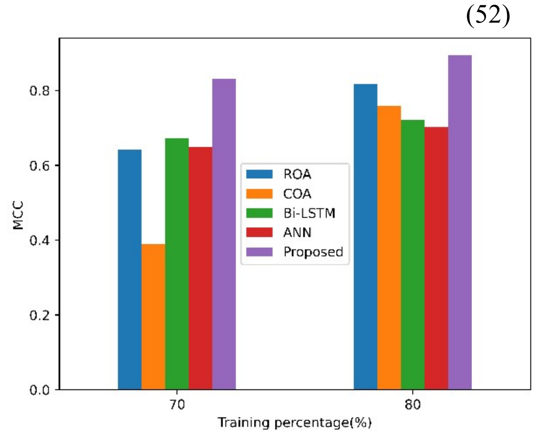

Figure 9: Evaluation of recommended and existing methods with regard to MCC

Figure 9 shows the performance of the recommended and approaches in use with regard to MCC. From the graph, the proposed

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

approach has a MCC of about 89% whereas the existing methods ROA, COA, Bi-LSTM and ANN has the MCC of 81%, 75%, 72% and 70% respectively. Thus, from the graphical representation, it is seen that the proposed approach has a higher MCC.

5.CONCLUSION

To overcome the challenges of various methods used for the video Anomaly Detection a method called Entropy based cascaded capsule neural network (CCNN) with an optimized LSTM (OLSTM) for segmentation and classification is proposed. In this method, pre-processing is done using the Gaussian filter followed by the segmentation using the Cascaded Capsule Neural network. Using this segmented data along with the dataset classified using entropy, feature extraction is done by which the features are extracted. Finally, classification is done using the Optimized LSTM using ROCO algorithm method. The suggested approach has an accuracy of 98%, specificity of 99.8%, sensitivity of 92.9%, Precision of 92.9, F measure of 92.9%, FNR of 10%, FPR of 0.3%and Matthew Correlation Coefficient (MCC) of 89%. Thus, from the results, it is seen that our suggested approach performs better in comparison to the other existing methods. This study has certain limitations and threats to validity. The availability of high-quality annotated datasets is limited, and dataset biases may affect model fairness. High computational costs and hardware dependencies impact scalability. Model generalizability is a concern due to overfitting and variations in imaging techniques. Extensive clinical validation is required, and reliance on retrospective data may limit real-time applicability. Ethical and regulatory challenges include compliance with data privacy laws and concerns regarding informed consent and data ownership in patient-specific digital twins. Enhancing medical imaging models requires diverse, well-annotated datasets for fairness and generalizability while mitigating biases. Optimizing deep learning models with lightweight architectures, pruning, and quantization improves efficiency and scalability. Robust models leveraging transfer learning and domain adaptation enhance generalization across imaging variations. Real-time clinical validation, through collaboration with healthcare institutions, ensures real-world effectiveness. Addressing ethical and regulatory challenges necessitates compliance with HIPAA and GDPR, along with considerations for informed consent and data ownership in patient-specific digital twins.

DECLARATIONS:

FUNDING

On Behalf of all authors the corresponding author states that they did not receive any funds for this project.

CONFLICTS OF INTEREST

The authors declare that we have no conflict of interest.

COMPETING INTERESTS

The authors declare that we have no competing interest.

DATA AVAILABILITY STATEMENT

All the data is collected from the simulation reports of the software and tools used by the authors. Authors are working on implementing the same using real world data with appropriate permissions.

ETHICS APPROVAL

No ethics approval is required.

CONSENT TO PARTICIPATE

Not Applicable

CONSENT FOR PUBLICATION

Not Applicable

REFERENCES

[1] Pang G, Shen C, Van Den Hengel A. Deep anomaly detection with deviation networks. InProceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining 2019 Jul 25 (pp. 353-362).

[2] Pang G, Shen C, Cao L, Hengel AV. Deep learning for anomaly detection: A review. ACM computing surveys (CSUR). 2021 Mar 5;54(2):1-38.

[3] Mehrotra KG, Mohan CK, Huang H, Mehrotra KG, Mohan CK, Huang H. Algorithms for time series data. Anomaly Detection Principles and Algorithms. 2017:153-89.

[4] Giorgi G, Abbasi W, Saracino A. Privacy-preserving analysis for remote video anomaly detection in real life environments.

[5] Zhou JT, Du J, Zhu H, Peng X, Liu Y, Goh RS. Anomalynet: An anomaly detection network for video surveillance. IEEE

Transactions on Information Forensics and Security. 2019 Feb 22;14(10):2537-50.
[6] Cook AA, Mısırlı G, Fan Z. Anomaly detection for IoT time-series data: A survey. IEEE Internet of Things Journal. 2019 Dec 6;7(7):6481-94.
[7] DeMedeiros K, Hendawi A, Alvarez M. A survey of AI-based anomaly detection in IoT and sensor networks. Sensors. 2023 Jan 25;23(3):1352.
[8] Khan SW, Hafeez Q, Khalid MI, Alroobaea R, Hussain S, Iqbal J, Almotiri J, Ullah SS. Anomaly detection in traffic surveillance videos using deep learning. Sensors. 2022 Aug 31;22(17):6563.
[9] Chen Z, Yeo CK, Lee BS, Lau CT. Autoencoder-based network anomaly detection. In2018 Wireless telecommunications symposium (WTS) 2018 Apr 17 (pp. 1-5). IEEE.
[10] Patrikar DR, Parate MR. Anomaly detection using edge computing in video surveillance system. International Journal of Multimedia Information Retrieval. 2022 Jun;11(2):85-110.
[11] Sultani W, Chen C, Shah M. Real-world anomaly detection in surveillance videos. InProceedings of the IEEE conference on computer vision and pattern recognition 2018 (pp. 6479-6488).
[12] Anoopa S, Salim AJ. Survey on anomaly detection in surveillance videos. Materials Today: Proceedings. 2022 Jan 1; 58:162-7.
[13] Şengönül E, Samet R, Abu Al-Haija Q, Alqahtani A, Alturki B, Alsulami AA. An analysis of artificial intelligence techniques in surveillance video anomaly detection: A comprehensive survey. Applied Sciences. 2023 Apr 14;13(8):4956.
[14] Thudumu S, Branch P, Jin J, Singh J. A comprehensive survey of anomaly detection techniques for high dimensional big data. Journal of big data. 2020 Dec; 7:1-30.
[15] Himeur Y, Ghanem K, Alsalemi A, Bensaali F, Amira A. Artificial intelligence-based anomaly detection of energy consumption in buildings: A review, current trends and new perspectives. Applied Energy. 2021 Apr 1; 287:116601.
[16] Chang, K., Yoo, Y. and Baek, J.G., 2021. Anomaly detection using signal segmentation and one-class classification in diffusion process of semiconductor manufacturing. Sensors, 21(11), p.3880.
[17] Chan, R., Lis, K., Uhlemeyer, S., Blum, H., Honari, S., Siegwart, R., Fua, P., Salzmann, M. and Rottmann, M., 2021. Segmentmeifyoucan: A benchmark for anomaly segmentation. arXiv preprint arXiv:2104.14812.
[18] Liu, F., Zhou, X., Cao, J., Wang, Z., Wang, T., Wang, H. and Zhang, Y., 2020. Anomaly detection in quasi-periodic time series based on automatic data segmentation and attentional LSTM-CNN. IEEE Transactions on Knowledge and Data Engineering, 34(6), pp.2626-2640.
[19] Hansen, S., Gautam, S., Jenssen, R. and Kampffmeyer, M., 2022. Anomaly detection-inspired few-shot medical image segmentation through self-supervision with supervoxels. Medical Image Analysis, 78, p.102385.
[20] Hassan, T., Akçay, S., Bennamoun, M., Khan, S. and Werghi, N., 2023. Unsupervised anomaly instance segmentation for baggage threat recognition. Journal of Ambient Intelligence and Humanized Computing, pp.1-12.
[21] Dissanayake, T., Fernando, T., Denman, S., Sridharan, S., Ghaemmaghami, H. and Fookes, C., 2020. A robust interpretable deep learning classifier for heart anomaly detection without segmentation. IEEE Journal of Biomedical and Health Informatics, 25(6), pp.2162-2171.
[22] Shi, Y., Yang, J. and Qi, Z., 2021. Unsupervised anomaly segmentation via deep feature reconstruction. Neurocomputing, 424, pp.9-22.
[23] Coelho, D., Costa, D., Rocha, E.M., Almeida, D. and Santos, J.P., 2022. Predictive maintenance on sensorized stamping presses by time series segmentation, anomaly detection, and classification algorithms. Procedia Computer Science, 200, pp.1184-1193.
[24] Song, Y., Xia, W., Li, Y., Li, H., Yuan, M. and Zhang, Q., 2024. Anomalyseg: Deep learning-based fast anomaly segmentation approach for surface defect detection. Electronics, 13(2), p.284.
[25] Wei, S., Wei, X., Ma, Z., Dong, S., Zhang, S. and Gong, Y., 2024. Few-shot online anomaly detection and segmentation.

Knowledge-Based Systems, 300, p.112168.
[26] Talaat FM, Gamel SA. RL based hyper-parameters optimization algorithm (ROA) for convolutional neural network. Journal of Ambient Intelligence and Humanized Computing. 2023 Oct;14(10):13349-59.
[27] Jebril NA, Abu Al-Haija Q. Cuckoo optimization algorithm (COA) for image processing. Nature Inspired Optimization Techniques for Image Processing Applications. 2019:189-213.
[28] Crisóstomo de Castro Filho H, Abílio de Carvalho Júnior O, Ferreira de Carvalho OL, Pozzobon de Bem P, dos Santos de Moura R, Olino de Albuquerque A, Rosa Silva C, Guimaraes Ferreira PH, Fontes Guimarães R, Trancoso Gomes RA. Rice crop detection using LSTM, Bi-LSTM, and machine learning models from Sentinel-1 time series. Remote Sensing. 2020 Aug 18;12(16):2655.
[29] Azgomi H, Haredasht FR, Motlagh MR. Diagnosis of some apple fruit diseases by using image processing and artificial neural network. Food Control. 2023 Mar 1; 145:109484.