.

CHAPTER 4

RESNET - LSTM APPROACH FOR EFFECTIVE VIDEO ANOMALY DETECTION IN SURVEILLANCE SYSTEMS

4.1 INTRODUCTION

The traditional Video Anomaly Detection (VAD) methods struggle with limited modeling capabilities and complex video relationships capturing capacity. Even though the CNN-YOLO model discussed in Chapter 3 achieved high-performance metrics but processes only 100 image frames and process only a single random frame, making it less reliable for continuous video evaluation with limited scalability. So proposed a hybrid approach which combines Residual Network-50 (ResNet-50) and Long Short-Term Memory (LSTM) for efficient detection of anomalous activities that processes more than 1000 frames for sequential video analysis.

4.2 RESNET- 50 ARCHITECTURE

A Deep Convolutional Neural Network (CNN) model ResNet-50, which is intended for image classification utilizes a residual learning framework for enhanced training efficiency. The architecture encompasses 50 layers, including 48 convolutional layers, a Max Pooling layer and an Average Pooling layer (Mandal et al., 2021). The ResNet-50 architecture is given in Figure 4.1.

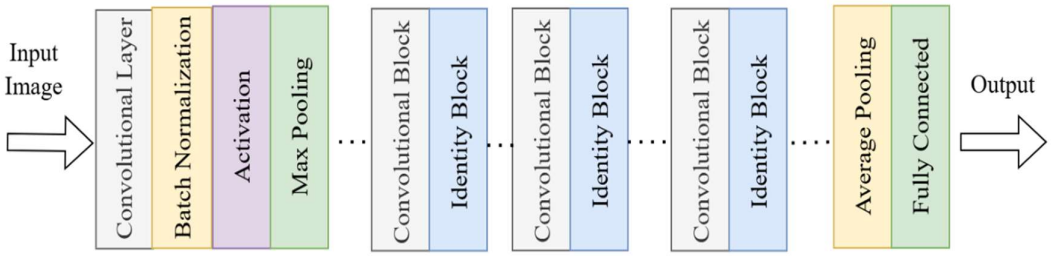

Figure 4.1 ResNet-50 Architecture

The ResNet-50 has four primary segments: convolutional layers, identity blocks, convolutional blocks and fully connected layers. The first convolutional layer extracts the features from the input image, then identity and convolutional blocks refine and modify these extracted features. The final fully interconnected layers generate the output classification.

During the feature extraction, the input image is processed by the ResNet-50 architecture, where convolutional layers, followed by batch normalization and Rectified Linear Unit (ReLU) activation, extract relevant visual features such as edges, textures and shapes.

To reduce the spatial dimensions and for preserving relevant features, the output from the activation layer is fed to the Max pooling layer. The identity and convolutional blocks, the major components of ResNet-50, are imprinted in Figures 4.2 and 4.3.

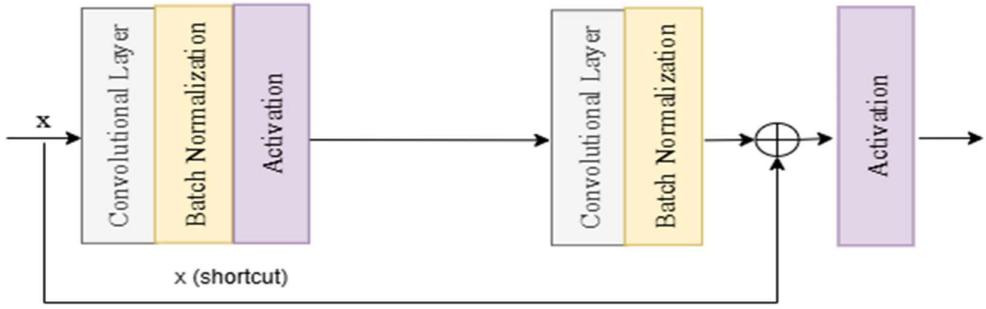

Figure 4.2 Identity Block

The input to the identity blocks is passed through a sequence of convolutional layers, incorporating skip connections to maintain original input and provide efficient residual learning. Convolutional blocks, on the other hand, function similarly but introduce an additional 1×1 convolutional layer, which adjusts the amount of filters before applying the 3×3 convolution, ensuring dimensional consistency and effective down sampling.

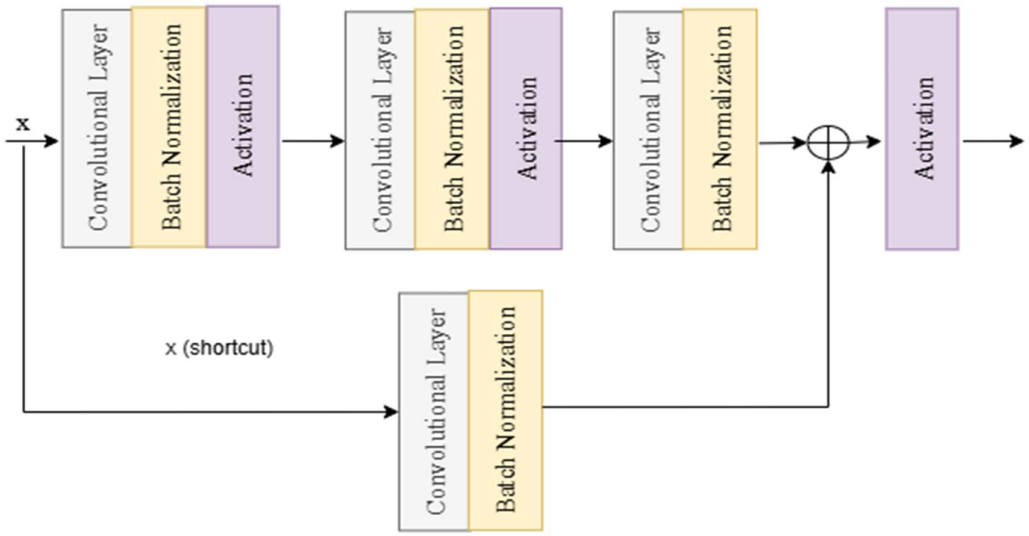

Figure 4.3 Convolutional Block

During these traversals, these layers refine feature representations, capturing deeper levels of abstraction. The final phase consists of fully connected layers that process high-level feature representations for classification. The output from the final fully connected layer passes through a Softmax activation function, producing class probability scores that allow the model to accurately categorize the input image.

- **Residual Learning and Skip Connections**

Deep neural networks often encounter the vanishing gradient problem, where extremely small gradient values delay effective weight updates during backpropagation, making training more challenging. ResNet-50 resolves this issue through residual blocks, enabling the model to learn residual functions instead of directly mapping the entire transformation. The residual learning block is provided in Figure 4.4.

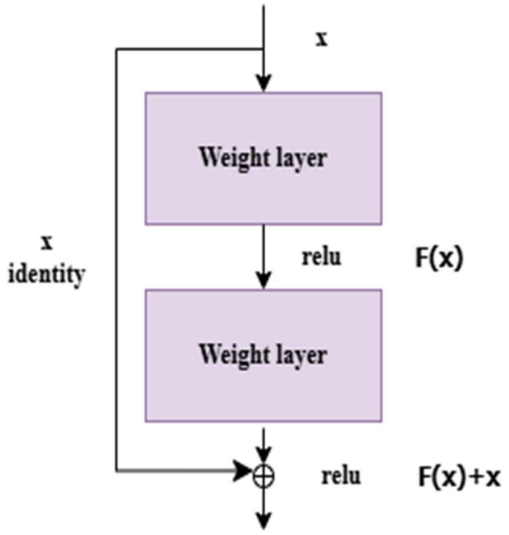

Figure 4.4 Residual Learning Block

The fundamental equation defining residual learning in ResNet-50 is provided in Equation 4.1:

$$H(x)=F(x)+x \tag{4.1}$$

where:

1. x (Input): The feature map from the previous layer, directly passed through a skip connection.
2. $F(x)$ (Residual Mapping): The transformation applied by convolutional layers, batch normalization and activation functions within the residual block.

3. $+x$ (Skip Connection): The identity mapping that retains the original input x and adds it to $F(x)$, ensuring smooth gradient flow and preventing degradation in deeper networks.

4. $H(x)$ (Final Output): The sum of x and $F(x)$, which serves as the result of the residual block and is forwarded to subsequent layers.

The residuals (difference) between the input and output than the entire transformation provide the efficiency of ReNet. The identity connection present in each block bypasses the intermediate layers reusing the earlier activation and maintain gradient flow and avoid vanishing gradient. The training stability is obtained using residual learning approach make the ResNet-50 to be applied in image recognition and feature extraction applications.

4.3 LONG SHORT-TERM MEMORY (LSTM) MODEL

In order to identify long-term dependencies in sequential data, a variant of Recurrent Neural Network (RNN) called LSTM is used. The vanishing or explored gradient issue faced in RNN is addressed in the LSTM by enabling the gates in the internal memory cells. Figure 4.5 demonstrates the architecture of the LSTM model.

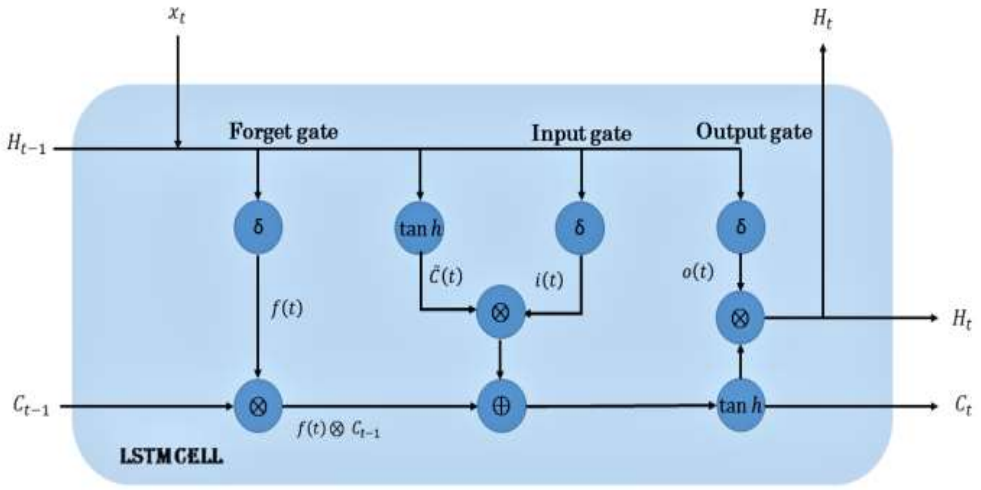

Figure 4.5 Architecture of LSTM Model (Qadeer et al., 2020)

The LSTM cell state, illustrated as a horizontal line running through the network, acts as a memory unit, enabling information to be retained or discarded through gates. The main role of the cell state is to preserve and transfer relevant information across time steps, ensuring that earlier data can persists throughout the sequence, thus addressing the Short-

term memory limitations of traditional recurrent networks. As processing continues, gates regulate the addition or removal of data from the cell state, determining what to be reserved or forgotten. The forget gate, the first in the sequence, evaluates necessary information to retain or discard using a sigmoid activation function defined in Equation (4.2).

$$sigmoid(x) = \frac{1}{1+e^{-x}} \tag{4.2}$$

The Sigmoid function provides values between zero and one, controlling the extent to which each element is passes through. Meanwhile, the tanh activation function yields a new candidate vector that contributes to the updated cell state. The updated state is derived depending upon the outputs from the gates, ensuring efficient memory retention.

The flow of data through the LSTM network is mathematically expressed in Equations (4.3) to (4.8):

1. **Forget Gate:** Determines either to retain or discard data from the preceding cell state.

$$f_t = \sigma\,(W_f \,.\, [h_{t-1}, x_t] + b_f) \tag{4.3}$$

where,

— f_t is the forget gate activation.

— h_{t-1} is the preceding hidden state.

— x_t is the existing input at time *t*.

— W_f is the weight matrix and b_f is the bias associated with the forget gate.

— σ is the sigmoid activation function.

2. **Input Gate:** Regulates the new data added to the cell state.

$$i_t = \sigma\,(W_i \,.\, [h_{t-1}, x_t] + b_i) \tag{4.4}$$

where,

— i_t is the input gate activation.

— W_i is the weight matrix and b_i the bias function associated with the input gate.

3. **Candidate Cell State:** Produces new candidate values to be added to the cell state.

$$\hat{C}_t = tanh\,(W_C \,.\, [h_{t-1}, x_t] + b_C) \tag{4.5}$$

where,

— $\hat{C}_t$ *is the candidate cell state.*

— W_c signifies the weight matrix and b_c represents the bias associated with the candidate state.

4. **Cell State Update:** Integrates the forget gate output and new candidate values to update the cell state.

$$C_t = f_t * C_{t-1} + i_t * \hat{C}_t) \quad (4.6)$$

where,

— C_t is the updated cell state.

— C_{t-1} is the previous cell state.

5. **Output Gate:** Controls the final output of the LSTM unit.

$$o_t = \sigma(W_o . [h_{t-1}, x_t] + b_o) \quad (4.7)$$

where,

— o_t is the output gate activation.

— W_o is the weight matrix and b_o is the bias term associated with the output gate.

6. **Hidden State Update:** Processes the final output based on the updated cell state.

$$h_t = o_t * tanh\ (C_t) \quad (4.8)$$

where,

— h_t is the final hidden state output of the LSTM unit.

— o_t is the output gate activation.

By leveraging gate mechanisms and cell state updates, LSTM networks efficiently retain long-term relationships, rendering them ideal for use in fields like time-series analysis, speech recognition and Natural Language Processing (NLP).

4.4 HYBRID RESNET-LSTM MODEL

Integrating ResNet-50 and LSTM provides a robust framework for efficient VAD by leveraging spatial feature extraction with temporal sequence modeling. Figure 4.6 represents the architecture of the ResNet – LSTM model for VAD, which incorporates of three key modules: preprocessing, feature extraction and classification.

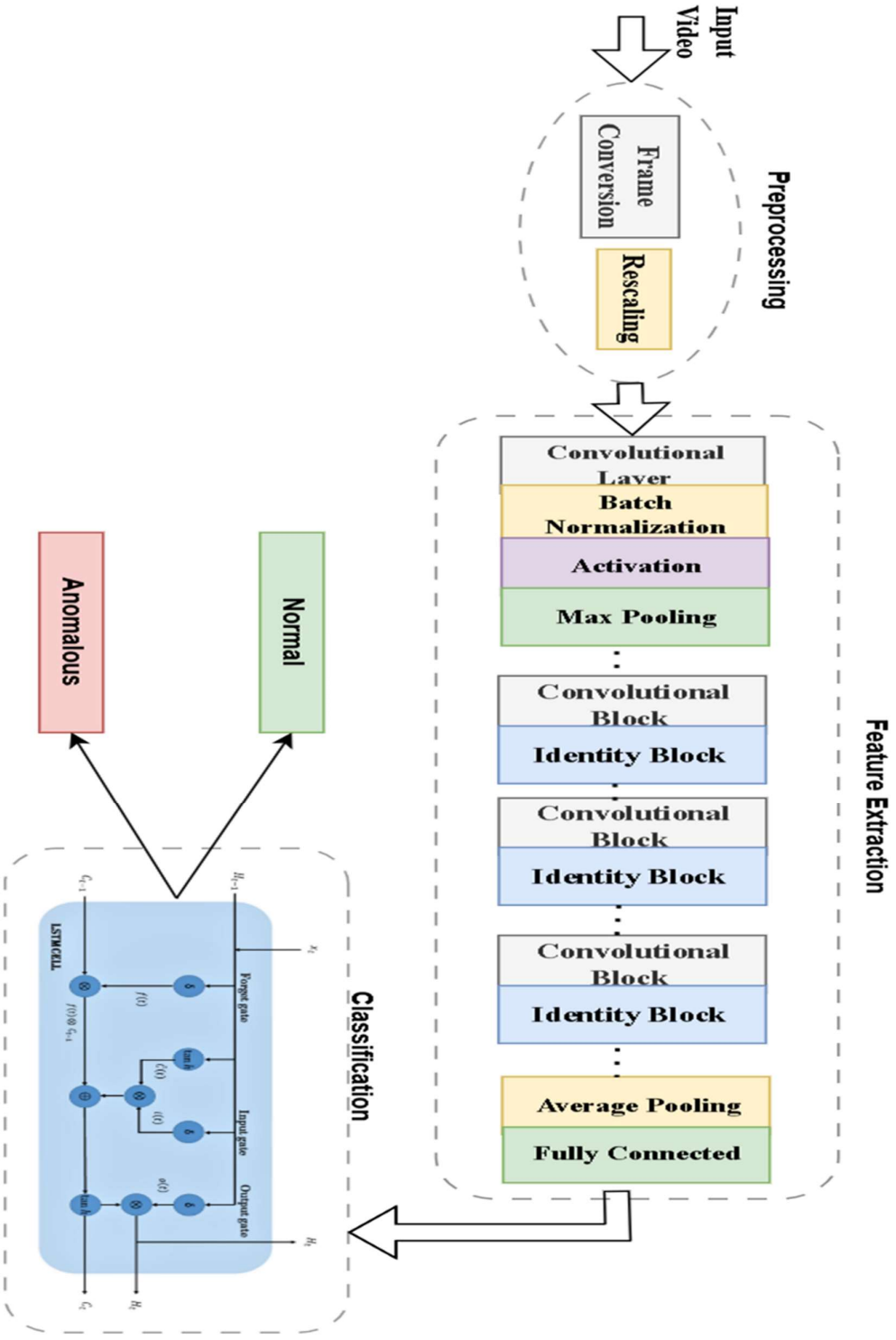

Figure 4.6 Architecture of Hybrid ResNet - LSTM model

The ResNet-LSTM model follows a structured pipeline for detecting anomalies in video data, leveraging spatial feature extraction using ResNet-50 and temporal modeling using LSTM. The training and validation sets are derived from the UCSD dataset, while the test set is utilized to evaluate the model's generalization capability. The hybrid ResNet-LSTM model comprises of preprocessing, feature extraction and classification stages.

- **Preprocessing**

The input video is initially segmented into individual frames, which are then resized and normalized to ensure consistency across samples. Pixel normalization or standardization is applied depending on the dataset characteristics:

— Normalization adjusts pixel values within a previously defined range (e.g., [0,1]), ensuring uniformity across all frames.

— Standardization adjusts pixel values to zero mean and unit variance, which benefits datasets with a broader distribution.

Additional preprocessing includes frame resizing, redundant frame removal and optional cropping or augmentation techniques to improve model robustness and reduce biases.

- **Feature Extraction**

Processing each video frame individually is computationally expensive due to redundancy and duplicate information. To improve efficiency, only the most relevant spatial features are extracted for effective learning.

A pretrained ResNet-50 model extracts a feature vector from each frame, retaining essential spatial attributes while discarding irrelevant details. The ReLU (Rectified Linear Unit) activation function boosts the network's capability to differentiate complex spatial patterns, ensuring a robust feature representation.

- **Classification**

After feature extraction, the series of feature vectors is passed to an LSTM network, which learns the temporal dependencies across frames. This allows the model to understand motion patterns and detect abnormal activity across time. The LSTM's gated memory mechanism retains only relevant information while filtering out less significant details. The LSTM network classifies video sequences based on their learned temporal dependencies.

The terminal classification layer utilizes a Softmax function to generate probability scores, classifying each frame into:

- Class 0 (Anomalous Activity)
- Class 1 (Normal Activity)

Algorithm 4.1 presents a pseudo-code for the Hybrid ResNet - LSTM model.

Algorithm 4.1: Pseudocode for Hybrid VAD using ResNet - LSTM

Step 1: Preprocessing

Load Video Frames

Read and load the video frames from the source video.

Rescaling

Frames are resized to match the feature extraction method's input dimensions.

Step 2: Feature Extraction with ResNet-50

Initialize ResNet-50 Model

Load a priorly trained ResNet-50 model to perform feature extraction.

Extract Features from Frames

Apply the ResNet-50 model to identify spatial patterns within every frame of the video.

Step 3: Temporal Feature Extraction with LSTM

Initialize LSTM Model

Define and initialize the LSTM model for temporal feature extraction.

Extract Temporal Features

Use the LSTM model to extract temporal relationships in the extracted features.

Step 4: Classification

Classify Anomalies

Flatten the LSTM output and pass it through Dense layers with Softmax activation for classification.

4.5 RESULTS AND DISCUSSIONS

The ResNet-LSTM model employs ResNet-50 for robust hierarchical feature extraction, efficiently capturing multiscale and intricate features. In addition, it incorporates LSTM units to effectively model temporal dependencies, thereby enriching its contextual understanding and overall performance. Figure 4.7 displays a normal image frame, representing typical pedestrian movement where people are walking naturally without any unusual activities.

Figure 4.7 VAD of Normal Image Frame

Figure 4.8 illustrates an example of an anomalous image frame, where a boy riding a bicycle and a vehicle entering the walkway deviate from the expected pedestrian behavior.

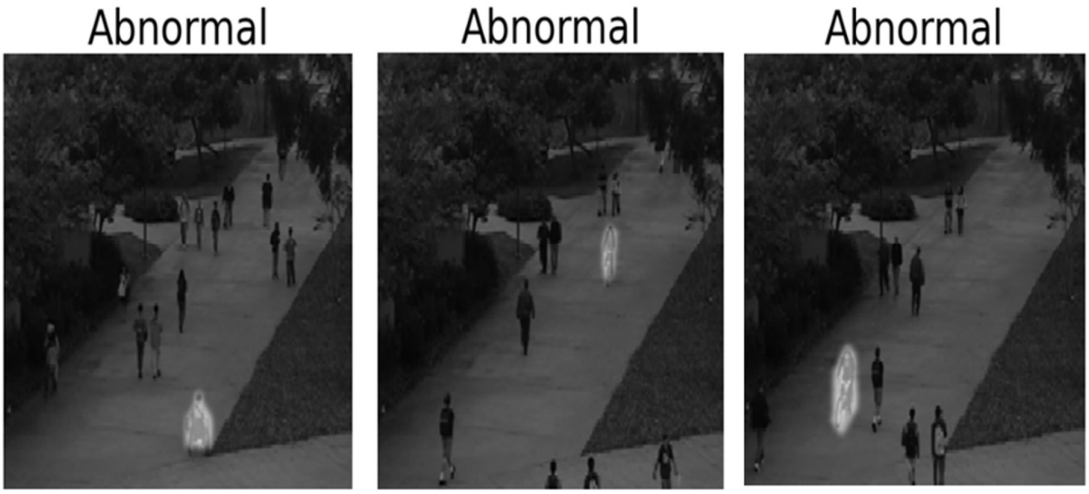

Figure 4.8 VAD of Anomalous Image Frame

4.5.1 Performance Evaluation

The efficiency of the hybrid ResNet-LSTM model in identifying video frames into two categories: normal and anomalous is examined in this section. Over 25 epochs of training, the model achieves notable accuracy and rapid convergence, as seen in its low loss rate near the 25th epoch.

Table 4.1 contains the performance evaluation for the ResNet-LSTM model, which include an Accuracy, Precision, Recall and F1 Score of 96.5%, an AUC of 0.99, a PSNR of 28.55dB and an EER of 14.5%.

Table 4.1 Performance of ResNet – LSTM Model

Performance Metrics	ResNet - LSTM
Accuracy	96.5 (%)
Precision	96.5 (%)
Recall	96.5 (%)
F1 Score	96.5 (%)
AUC	0.99
PSNR	28.55 (dB)
EER	14.5 (%)

Figures 4.9 and Figure 4.10 illustrate the training progress for accuracy and loss analysis, showing a greater accuracy level of 99.67% and a negligible loss level of 0.011% at the 25th epoch.

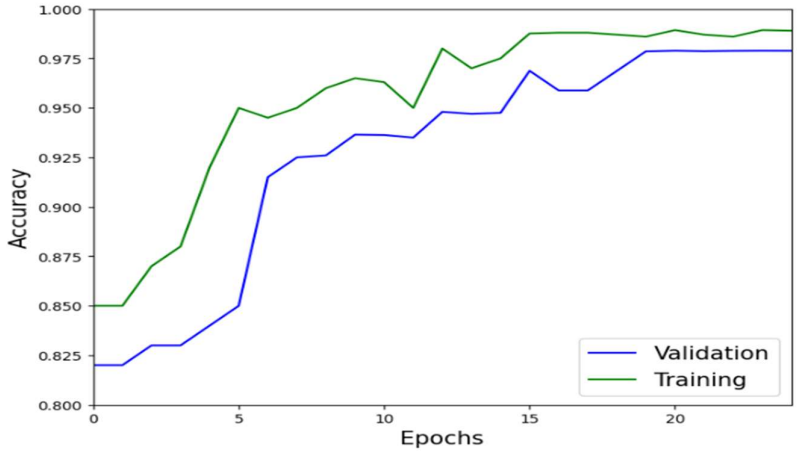

Figure 4.9 Training Progress of Accuracy Analysis

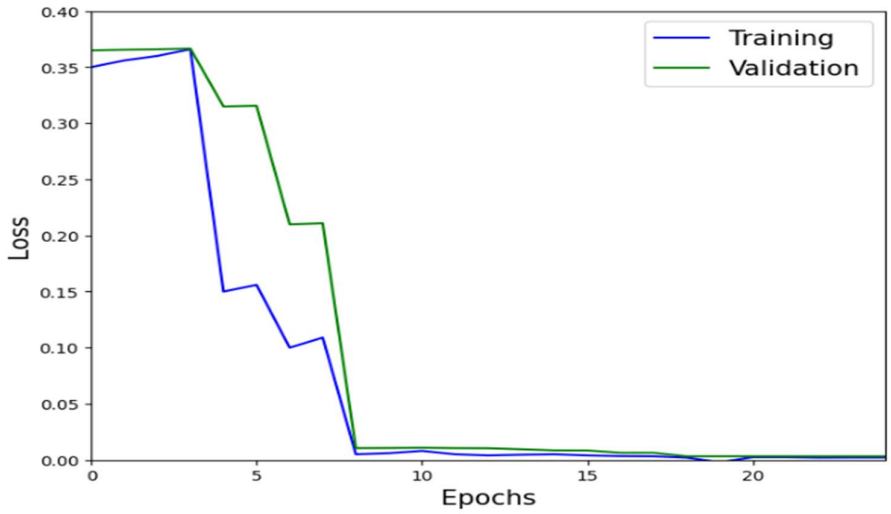

Figure 4.10 Training Progress of Loss Analysis

The performance of hybrid ResNet-LSTM model is analyzed with CNN-YOLO and the Hybrid Ensemble Recurrent Reinforcement (HERR) models [8] and the results are given in Figures 4.11 to 4.16. Figure 4.11 compares accuracy, showing that ResNet-LSTM achieves 96.5%, while HERR attains 91.5%. ResNet-LSTM outperforms HERR with a 5.46% improvement, demonstrating its superior classification capability. The lower accuracy of HERR suggests a higher likelihood of misclassification compared to ResNet-LSTM.

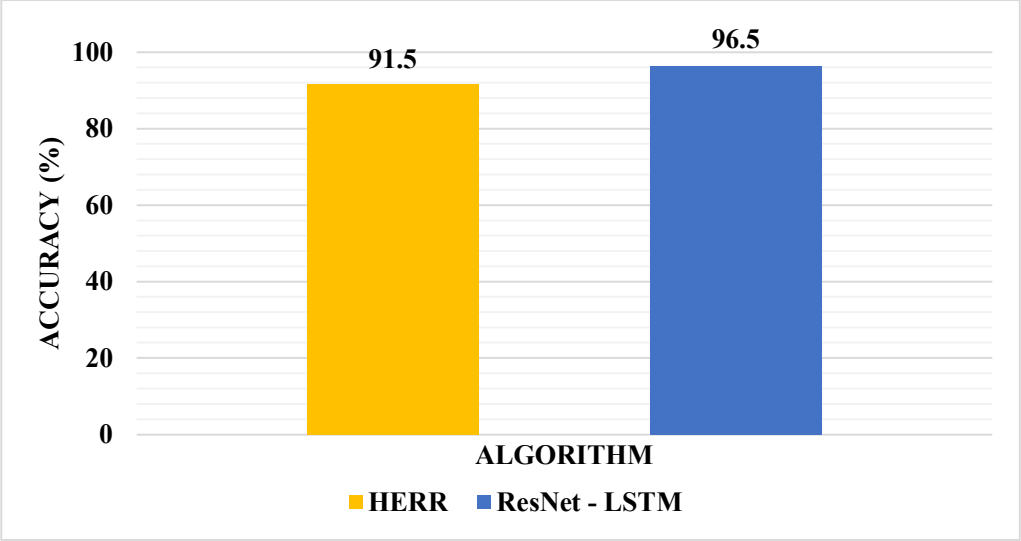

Figure 4.11 Performance Comparison of Accuracy

Figure 4.12 compares Precision, indicating that ResNet-LSTM achieves 96.5%, whereas HERR records 92%. ResNet-LSTM shows a 4.89% improvement over HERR, highlighting its better ability to minimize false positives. HERR, while performing well, shows a slightly reduced Precision, suggesting a higher tendency for incorrect positive classifications.

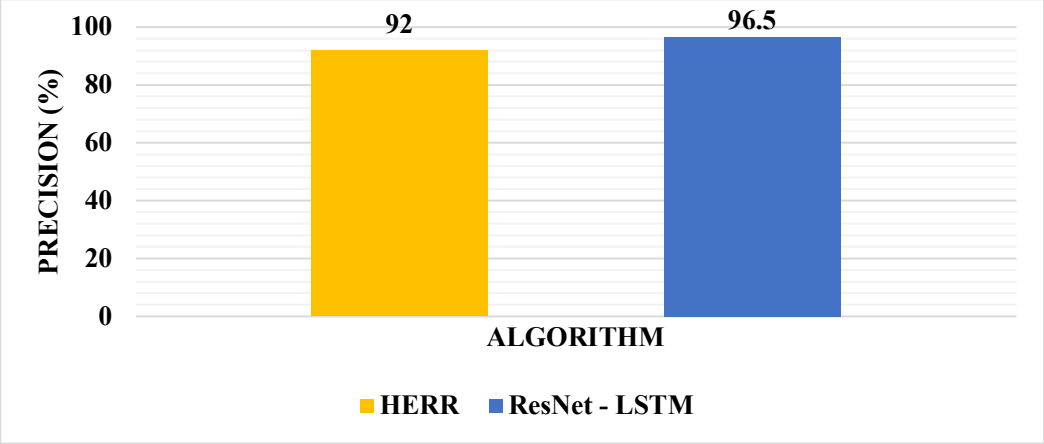

Figure 4.12 Performance Comparison of Precision

Figure 4.13 compares Recall, revealing that ResNet-LSTM reaches 96.5%, while HERR achieves 91.09%. ResNet-LSTM demonstrates a 5.94% improvement, indicating its stronger capability in detecting actual anomalies. HERR, though effective, has a lower Recall, implying a slightly higher rate of false negatives.

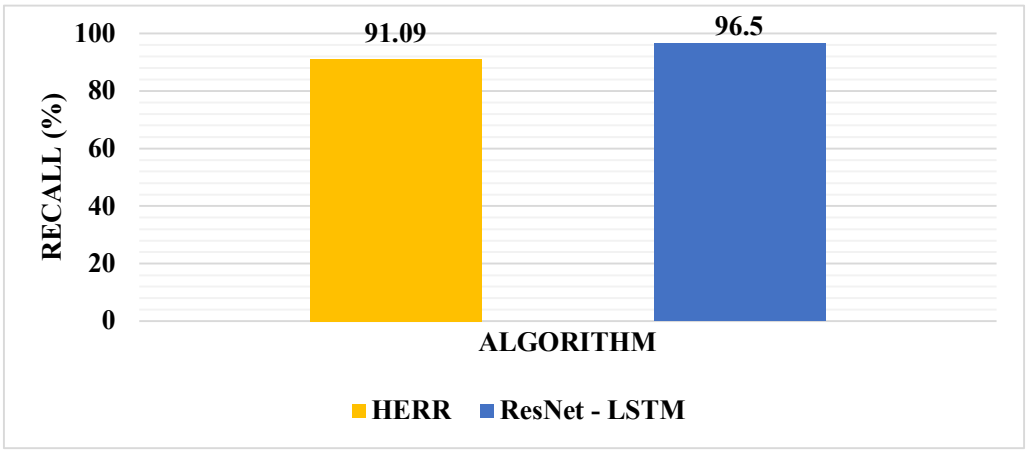

Figure 4.13 Performance Comparison of Recall

Figure 4.14 provides the performance comparison of F1 Score of ResNet-LSTM.

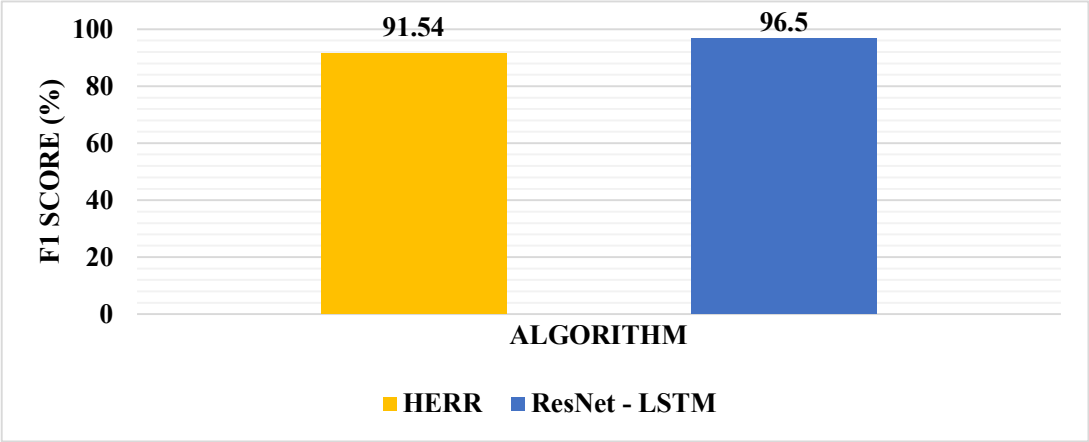

Figure 4.14 Performance Comparison of F1 Score

Based on the figure, the F1 Score shows that ResNet-LSTM achieves 96.5%, whereas HERR records 91.54%. ResNet-LSTM outperforms HERR by 5.42%, maintaining a superior trade-off between Precision and Recall. The reduced F1 Score of HERR suggests a slightly lower consistency in classification performance. Figure 4.15 provides the performance comparison of AUC of ResNet-LSTM.

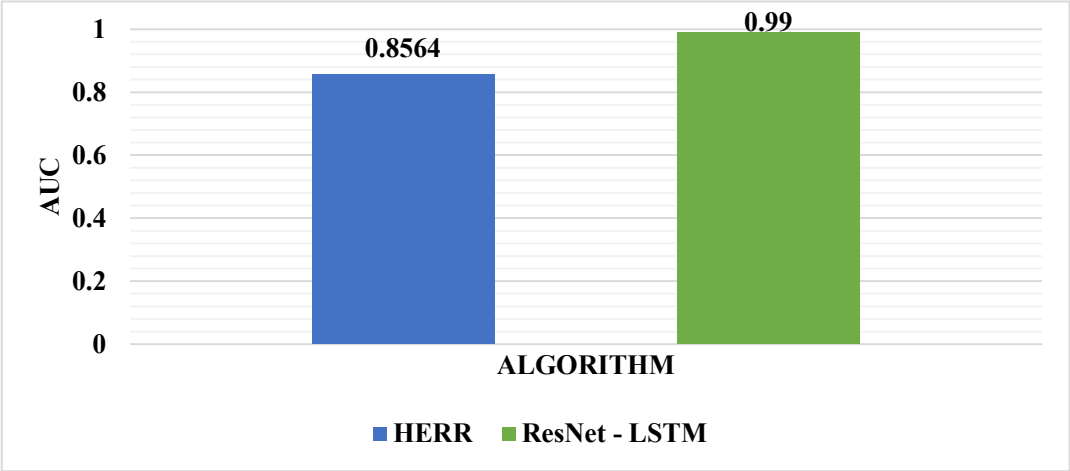

Figure 4.15 Performance Comparison of AUC

Figure 4.15 compares the AUC, demonstrating that ResNet-LSTM attains 99%, while HERR reaches 85.64%. ResNet-LSTM outperforms HERR by 15.57%, showcasing its superior capability to differentiate normal and anomalous instances. The lower AUC of HERR indicates reduced effectiveness in overall classification.

Figure 4.16 compares the EER, showing that ResNet-LSTM records 14.5%, while HERR achieves 14.33%. The minimal 1.17% difference suggests that both models perform similarly in balancing false positives and false negatives, with ResNet-LSTM having a slight advantage.

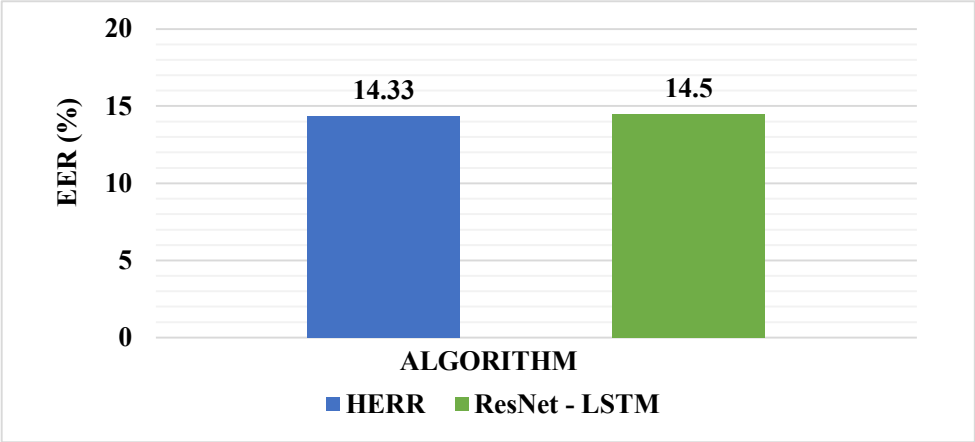

Figure 4.16 Performance Comparison of EER

The empirical findings demonstrate that the ResNet-LSTM model surpasses the HERR model in VAD. By effectively integrating spatial and temporal feature extraction, it achieves a strong balance between anomaly detection accuracy and image quality preservation. ResNet-50 efficiently extracts high-level features, capturing activity-discriminative information, while LSTM processes the feature frames to classify them as normal or anomalous. The capacity of the model to detect unusual activities is denoted with high Precision value. The efficiency of the model highlighted by the results enable the system to be applied in reliable and safe VAD application.

4.6 SUMMARY

The chapter presented a hybrid ResNET-LSTM model for VAD that utilized more than thousands of video frames and can analyze the sequential video information. The performance obtained is the model attained a notable classification accuracy of 96.5% that described the usefulness of the model. The metrics like Precision, Recall, F1 Score attained 96.5% and AUC of 0.99 which express the model's excellence in VAD. The model extracted feature from video input using a ResNet-50 and to capture the temporal relationship among video sequence and LSTM module is incorporated. The results obtained for the model express the potential of the hybrid model for enhanced VAD.