Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.055787

ARTICLE

# A Recurrent Neural Network for Multimodal Anomaly Detection by Using Spatio-Temporal Audio-Visual Data

**Sameema Tariq[1], Ata-Ur- Rehman[2,3], Maria Abubakar[2], Waseem Iqbal[4], Hatoon S. Alsagri[5], Yousef A. Alduraywish[5] and Haya Abdullah A. Alhakbani[5,*]**

[1]Department of Electrical Engineering, University of Engineering and Technology, Lahore, 54890, Pakistan

[2]Department of Electrical Engineering, National University of Science and Technology, National University of Sciences and Technology, Islamabad, 24090, Pakistan

[3]Department of Business and Computing, Ravensbourne University London, Ravensbourne University, London, SE10 0EW, England

[4]Electrical and Computer Engineering Department, College of Engineering, Sultan Qaboos University, Muscat, 123, Oman

[5]College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, 11673, Saudi Arabia

*Corresponding Author: Haya Abdullah A. Alhakbani. Email: hahakbani@imamu.edu.sa

Received: 07 July 2024 Accepted: 19 September 2024 Published: 18 November 2024

## ABSTRACT

In video surveillance, anomaly detection requires training machine learning models on spatio-temporal video sequences. However, sometimes the video-only data is not sufficient to accurately detect all the abnormal activities. Therefore, we propose a novel audio-visual spatiotemporal autoencoder specifically designed to detect anomalies for video surveillance by utilizing audio data along with video data. This paper presents a competitive approach to a multi-modal recurrent neural network for anomaly detection that combines separate spatial and temporal autoencoders to leverage both spatial and temporal features in audio-visual data. The proposed model is trained to produce low reconstruction error for normal data and high error for abnormal data, effectively distinguishing between the two and assigning an anomaly score. Training is conducted on normal datasets, while testing is performed on both normal and anomalous datasets. The anomaly scores from the models are combined using a late fusion technique, and a deep dense layer model is trained to produce decisive scores indicating whether a sequence is normal or anomalous. The model's performance is evaluated on the University of California, San Diego Pedestrian 2 (UCSD PED 2), University of Minnesota (UMN), and Tampere University of Technology (TUT) Rare Sound Events datasets using six evaluation metrics. It is compared with state-of-the-art methods depicting a high Area Under Curve (AUC) and a low Equal Error Rate (EER), achieving an (AUC) of 93.1 and an (EER) of 8.1 for the (UCSD) dataset, and an (AUC) of 94.9 and an (EER) of 5.9 for the UMN dataset. The evaluations demonstrate that the joint results from the combined audio-visual model outperform those from separate models, highlighting the competitive advantage of the proposed multi-modal approach.

## KEYWORDS

Acoustic-visual anomaly detection; sequence-to-sequence autoencoder; reconstruction error; late fusion; regularity score

**Nomenclature**

| | |
|---|---|
| AAD | Acoustic Anomaly Detection |
| ADAM | ADAptive Moment |
| AE | AutoEncoder |
| AUC | Area Under Curve |
| BCE | Binary Cross Entropy |
| CAE | Convolutional AutoEncoder |
| CCTV | Closed-Circuit Television |
| CNN | Convolutional Neural Network |
| CNN | Convolutional Neural Network |
| ConvLSTM | Convolutional Long Short-Term Memory |
| DAE | Denoising AutoEncoder |
| EER | Equal Error Rate |
| FNR | False Negative Rate |
| FP | False Positive |
| FPR | False Positive Rate |
| GRU | Gated Recurrent Unit |
| HOF | Histogram Optical Flow |
| HoG | Histograms of Gradient |
| IF | Isolation Forest |
| JAD | Joint Anomaly Detection |
| Leaky ReLU | Leaky Rectified Linear Unit |
| LSTM | Long Short-Term Memory |
| LSTMAE | Long Short-Term Memory AutoEncoder |
| MFCC | Mel-Frequency Cepstral Coefficients |
| MSE | Mean Square Error |
| OCC | One-Class Classification |
| OC-SVM | One-Class Support Vector Machine |
| PCA | Principal Component Analysis |
| PSO | Particle Swarm Optimization |
| ReLU | Rectified Linear Unit |
| SAE | Sparse AutoEncoder |
| SC | Spectral Centroid |
| SFM | Social Force Model |
| STFT | Short-Term Fourier Transform |
| SVM | Support Vector Machine |
| TNR | True Negative Rate |
| TPR | True Positive Rate |
| TUT | Tampere University of Technology |
| UCSD PED 2 | University of California, San Diego Pedestrian 2 |
| UMN | University of Minnesota |
| VAD | Visual Anomaly Detection |
| VAE | Variational AutoEncoder |

## 1 Introduction

Audio-visual surveillance is the process of analyzing audio and video for security purposes. It's a thought-provoking topic that deals with audio signal processing and the computer vision approach. Anomaly detection is an important part of this process. Stated goals for ideal anomaly detection include gradual online adaptive tracking of normal events and abrupt real-time recording of abnormal events [1]. Falling, entering a prohibited location, accidents, gunshots, yelling, criminal behavior, and many more are examples of anomalous events [2].

Anomalous Event Detection is complicated because data for this type of event is infrequent, the data that is collected is noisy, data gathering for this type of event is dangerous, and the definition of an anomalous event differs depending on the application. For instance, people run on roads to define an anomalous event, but people run on football grounds to define a normal event.

The motivation for this approach is based on the fact that audio and video models complement each other. Visual and audio anomalies have been detected independently for years using a variety of traditional and modern machine learning methods. However, visual event detection frequently fails due to inaccuracy in distinguishing between distinct activities. In situations where visual details are ambiguous, audio information provides precise details. Similarly, when audio event detection fails to provide sufficient and error-free information, visual details are taken into account to complete the task [3].

The ideal functionality of an anomaly detection system involves two basic goals: first, it must be able to continually monitor and respond to what is deemed typical behavior or occurrences, learning and modifying over time as conditions change. This guarantees that the system is accurate and trustworthy in the long run. Second, it must be able to respond quickly when anything unusual occurs, capturing these occurrences in real time for further study or urgent action.

For example, illegally unlocking a door via (CCTV) appeared normal, but immediately sounding alarm aids in detecting anomalous behavior. Audio and video complement one another. Occlusion occurs in video, and temporal overlapping of events occurs in audio. The dark staircases, tunnels, and corners do not aid in video event detection, but any unusual sound can aid in the detection of abnormal events. As a result, the combined results of audio and video are far better compared to the individual results as depicted in Table 1.

**Table 1:** The table shows the (AUC) (%age) for audio, image, and combined audio-image data, demonstrating greater performance with combined data

| Kumari et al. | Audio | Image | Audio-image |
|---|---|---|---|
| (AUC) (%) | 50.39 | 75.82 | 76.44 |

By incorporating the strengths of both modalities, these structures provide a more complex and accurate knowledge of events, allowing for improved decision-making and increased overall security. As technology advances, the development of these audio-visual models is anticipated to result in more complex and effective anomaly detection systems. These developments not only strengthen security measures but also offer new avenues for their application in a variety of disciplines, including public and law enforcement, smart cities, and beyond. The future of surveillance resides in this comprehensive approach, in which audio and visual data work together to build safer, more responsive surroundings.

The proposed approach employs two data sources: audio and video. In the proposed method, deep learning and handcrafted approaches are used to extract features from both sources to learn spatial and temporal aspects. These features are fed into the deep learning autoencoder, which trains the model on normal events and tests it on aberrant ones. Normal events have a lower reconstruction error than abnormal occurrences. The outcomes of both sources are merged to forecast how likely the deep learning model is to predict frame abnormality based on audio and video outcomes. The contribution includes the development of the audio files against (UCSD PED 2) and (UMN) datasets. Along with this, acoustic-visual surveillance is introduced by designing a joint anomaly detection algorithm that outperforms individual algorithms.

There are certain challenges and limitations to the proposed research paper. Unfortunately, there is no publicly available audio-video dataset to evaluate the reliability of the system outside (UMN) and (UCSD). Furthermore, no research study has been done on both the audio-video (UMN) and (UCSD), preventing us from doing a comparison analysis that would have provided significant perspectives into the strengths and weaknesses of the systems. The advantages of the proposed research paper include a diversity of available datasets. The (UCSD) dataset is taken at different angles, while the (UMN) dataset comprises three different lighting conditions, proving the applicability and reactivity of the system. The contribution of the proposed approach includes the collection of anomalous and normal audio data from various YouTube and Google sources. This audio collection includes sounds such as carts, wheelchairs, skaters, bikes, gunshots, and crowd noise. These audio files have been synchronized with the (UCSD) and (UMN) video datasets.

The paper is organized as given: Section 2 describes the literature about each anomaly detection model: acoustic, visual, and joint. Section 3 discusses the methodology of the proposed solution and the anomaly detection techniques. Section 4 describes the experimental results, benchmark datasets, model parameters, and implementation details. Section 5 concludes the whole discussion about techniques employed for anomaly detection.

## 2 Literature Review

The literature review in the paper is divided into three subsections: acoustic anomaly detection, visual anomaly detection, and joint anomaly detection approaches.

### *2.1 Acoustic Anomaly Detection (AAD)*

Pereira et al. [4] proposed an in-vehicle unsupervised (AAD) system as part of a wider in-vehicle intelligence R&D initiative. The researchers initially created a new synthetic in-vehicle (AAD) simulator to develop three audio mixes containing background trip noises mixed with five normal and three abnormal occurrences. Subsequently, to execute (AAD), two sound feature extraction approaches were investigated, as well as a proposed (LSTMAE) method. Pooyan et al. [5] proposed the anomaly detection approach for the compression system using both the technique (OCC) and spectral. This research detects midstream compressor failures using audio sensor data. Initially, the input audio signals are used to construct (STFT), (MFCC), and (SC) characteristics. Second, to generate high-level features, deep learning-based feature extraction is used. Finally, normal and anomalous audio signals are classified using a (PCA) step and a (SVM). The suggested approach was tested on two datasets, including 10196 audio signals recorded from a compressor. A deep learning approach (AE) [6] is employed for (AAD) in various forms, like (DAE) [7], (SAE) [8], (CAE) [9], and (VAE) [10]. It trains the model to compress and reconstruct the normal instances, whereas abnormal instances are not reconstructed by the model. Kumari et al. [11] proposed a method for using Huffman coding. This method was identified to improve results with minimum processing overhead and is used for anomaly identification in audio to get characteristics such as flexible event length and less reliance on cluster data. Several studies have tackled (AAD) using various ML algorithms in recent years, including the (IF) [12] and the IRESE method [13]. Mnasri et al. [14] presented a novel method of anomaly detection comprised of (VAE) and interval-valued fuzzy sets. The proposed method integrates two approaches, which are autoregressive (VAEs) and interval-valued fuzzy sets. A probabilistic interval comparison method is utilized for defuzzification that detects the corresponding class. The dataset of this approach is road traffic surveillance that contains hazardous events, for instance, vehicle accidents using auditory signals. This proposed study concentrated on autoencoders (AE), a deep learning neural architecture that has gained popularity in the treatment of (AAD) [15]. (AE) training is computationally quicker than (OC-SVM), (IF), and IRESE; therefore, it can handle greater quantities of training data.

### *2.2 Visual Anomaly Detection (VAD)*

Pang et al. [16] demonstrated that using self-trained deep ordinal regression to detect video anomalies overcomes two main shortcomings of prior methods: reliance on manually labeled normal training material and suboptimal feature learning. An end-to-end trainable video anomaly detection technique is designed that enables integrated representation learning and anomaly scoring without manually annotating normal or abnormal data by developing a synthetic two-class ordinal regression task. Kumari et al. [17] developed a masking approach for enhancing resilience against background noise utilizing discriminators' class activation maps. It is a self-supervised masking framework that aims to picture discriminative regions to allow robust anomaly detection. The results show that in adversarial training, the discriminator's class activation map changes in three stages before settling on the foreground location in the pictures. These activation map characteristics create a mask that suppresses false signals from the background, allowing for robust anomaly identification by participating in local discriminative properties. Morais et al. [18] considered human detection and tracking reconstruction techniques. With the help of Alpha Pose [19], a constant length of tracks is approximated for skeletons that are fragmented into local and global modules. A two-branch framework with three (GRU)s [20]: an encoder, a reconstructing decoder, and a predicting decoder are proposed [21]. (VAD) is also built upon dictionary learning [22] technique. It learns a vocabulary of typical occurrences and detects the events that the dictionary cannot adequately express. Dictionary learning may also be used to learn low-level features like (HoG) or (HoF) [23], as well as 3D gradient features [24]. Other approaches, such as hashing-based methods [25] and clustering [26], have been developed to model normal events with compact representations. The most common techniques utilized currently include (CNN), (AEs), (LSTMs), and many more algorithms. Naud et al. [27] proposed a revolutionary hyperspherical (VAE) using stereographic projections with a gyroplane layer through theoretical and practical studies of manifold forms an equivalent to Poincare (VAE). It is unsupervised visual anomaly detection for embedding data distributions in constant curvature manifolds, which is advantageous in terms of model generalization and can result in better interpretable representations. The proposed approach employed the techniques of deep learning that are hybrid.

### *2.3 Joint Anomaly Detection (JAD)*

Wu et al. [28] proposed relational network-based multimodality audio-visual violence detection algorithms. A weak supervision neural network with three parallel branches captures different relations among video snippets and integrates features, where the holistic branch captures long-range dependencies using similarity prior, the localized branch captures local positional relations using proximity prior, and the scoring branch dynamically captures predicted score closeness. Kumari et al. [17] proposed an unsupervised multimodal anomaly detection approach for long-term surveillance based on the concept drift idea. The audio and video data are integrated and trained using a deep learning-based teacher-student network. Using principal component analysis, features from both inputs are merged and compressed. As a result, a teacher-student network is applied to these compressed characteristics to provide a shared representation of data. A multivariate adaptive Gaussian mixture model is used to learn the data dynamics. Rehman et al. [29] employed late fusion-based audio-visual anomaly detection for general monitoring. To identify anomalous video frames in the video modality, the optical flow was integrated with (PSO) and the (SFM). The (PSO) approach controls a swarm of data iteratively in conjunction with optical flow to determine the flow of moving objects in the crowd. Furthermore, (SFM) is used to quantify interaction forces among people in a crowd to define the population's behavior. The acoustic features-based (SVM) classifier is used to detect abnormalities in the audio modality. In addition, a late fusion is used to make the ultimate conclusion. Furthermore, early and late fusion techniques have been examined in these publications. Early fusion-based models [30] employ a simple assumption of conditional independence across the modalities, which may not hold in actuality. Late fusion, on the other hand, focuses on classifier prediction. As a result, researchers have embraced fusion-based techniques. Deep learning's undeniable success in modeling complex problems has compelled researchers to create deep learning-based fusion frameworks for a wide range of tasks, including activity detection [31], face recognition [32], multisource image pixel-wise classification [33], panchromatic and multispectral imagery classification [34], and so on. These frameworks offer significant benefits over traditional fusion approaches [35]. Imran et al. [36] offered behavioral-based database intrusion detection, which is appropriate for insider assaults, concentrating on feature selection and algorithm choice. Multimodal anomaly detection can be enhanced by implementing the above approach to detect sequences of malicious activities rather than isolated events. Spatiotemporal audio-visual data integration with these tactics can enhance complicated anomaly detection.

### *2.4 Limitations*

Pereira et al. [4] extracted two sound features to predict anomalies in their (AAD) system. If these techniques fail to capture the relevant features for effective anomaly detection, the system's performance might be compromised. Conversely, our model utilizes eight different acoustic features, providing a more detailed understanding of sound characteristics, which enhances its ability to predict anomalous features more accurately. Kumari et al. [11] suggested employing class activation maps to construct masks; however, these maps can be difficult to read, making it difficult to understand why some regions are marked as anomalous. In contrast, our model uses an autoencoder (AE) approach, which is simpler to grasp and comprehend because it provides a low reconstruction error for abnormal frames and vice versa. Wu et al. [28] implemented a weak supervision strategy, which can degrade model performance if the labels are noisy or erroneous. In contrast, our model employs an unsupervised approach, allowing it to independently learn and detect abnormalities. Rehman et al. [29] limited their method to a self-curated dataset with one sort of anomalous example, a gunshot, while our data is tested on six different sounds.

## 3 Methodology

The proposed approach is an unsupervised methodology based on the objective of predicting anomalous frames. This approach employs the concept of training an end-to-end model comprising

Sequence to Sequence AutoEncoder (Seq2Seq AE) as described in [1] and [2] to learn spatiotemporal features of multi-data. The block diagram of the proposed approach is shown in Fig. 1. It depicts that the proposed model is comprised of three sub-models:

- Acoustic Model
- Visual Model
- Joint Model

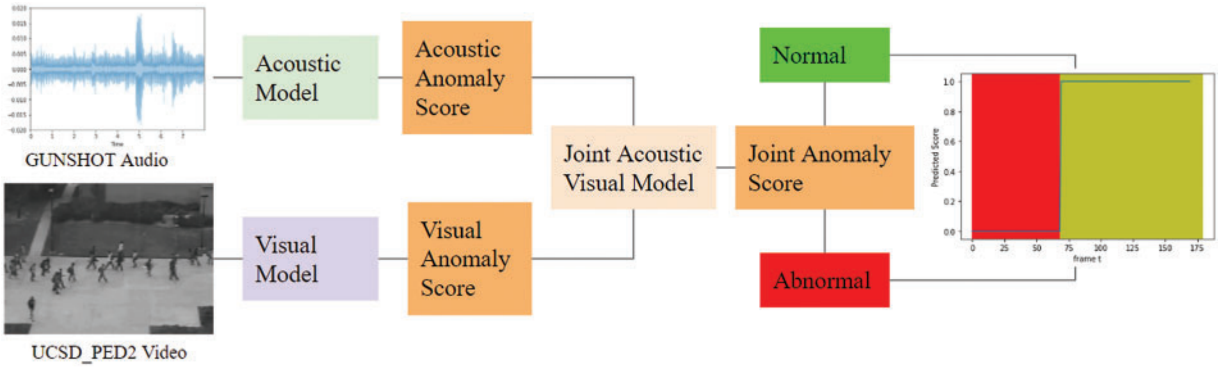


**Figure 1:** Block diagram: The proposed joint model computes audio and video inputs through acoustic and visual models to calculate anomaly scores

These scores are concatenated to train the joint model, classifying sequences as normal or abnormal based on a cutoff value between 0.7 and 0.92. Scores below the cutoff indicate anomalies, while those above are considered normal.

## *3.1 Acoustic Model*

### *3.1.1 Feature Extraction*

The audio signal is continuous, which is converted to a discrete format composed of sample data and a sample rate of 22 kHz. The proposed method processed this sample data to extract low-level features and analyze both spectral and time-domain features. The extracted features are Spectral Flatness [37], Zero Crossing Rate [38], Spectral Centroid [38], Spectral Bandwidth [38], (MFCCs) [39], Mel Spectrogram [39], Spectral Roll off [39], and Spectral Contrast [39]. These features are normalized, giving the input data shape as (audio_files, frame_sequence, feature_length).

### *3.1.2 Model Architecture*

Fig. 2 interprets the proposed acoustic model, while Table 2 shows the model architecture. The proposed acoustic model is comprised of Seq2Seq (AE) built upon 1-D (CNN) and (LSTM). 1-D (CNN) extracts higher-level features, while (LSTM) extracts temporal features. The proposed acoustic model employed Keras layers. Initially, a basic model is constructed via the sequential layer, and subsequent layers are connected to the model in sequence. These subsequent layers are comprised of one input layer, ten hidden layers, and one output layer. These hidden layers perform the encoding and decoding operations upon input. The processed acoustic input is fed into the deep learning model that contains filter, kernel, padding, stride, and activation layers. Each layer has a kernel size of 5 to extract the requisite details from each feature. The padding of the same is applied to ensure the output size is identical to the input as well as the filter processed upon all the elements.

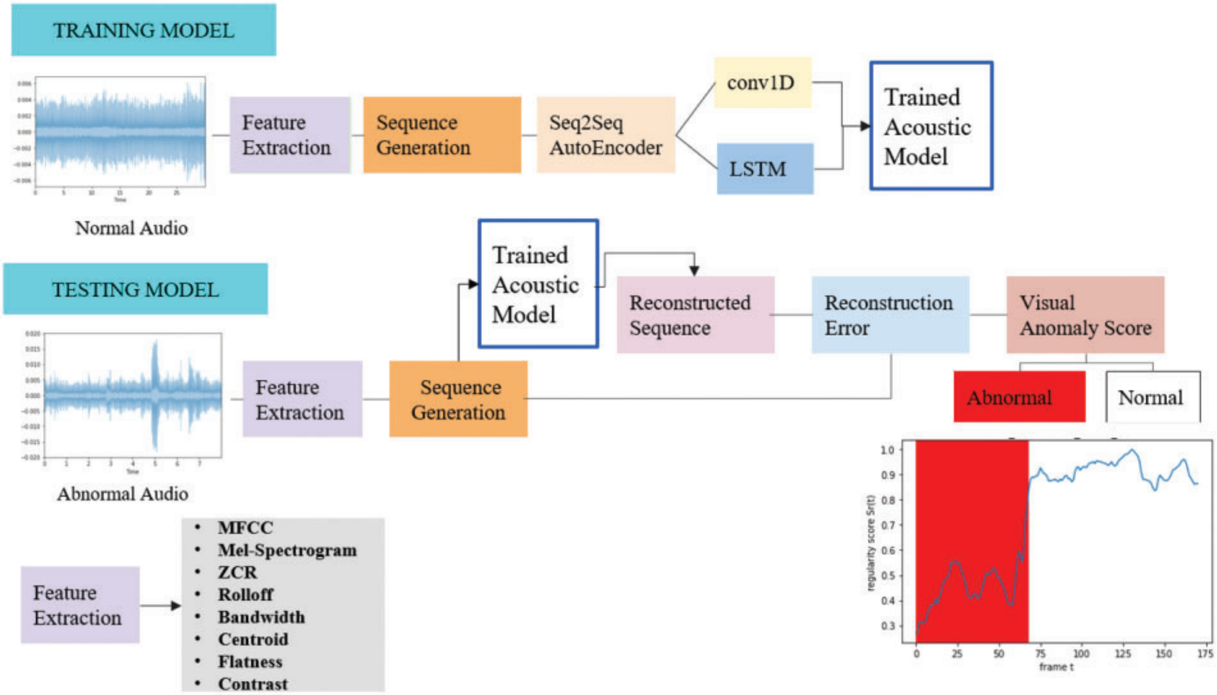


**Figure 2:** Block diagram: The proposed audio model processed audio signals to generate features that are trained upon Seq2Seq AutoEncoder, which uses a loss function and an optimizer. This trained model predicts test data by reconstructing sequences to originate the audio anomaly score

**Table 2:** Architecture: The table illustrates the neural network architecture for (AAD) which includes a Conv1D encoder, an LSTM encoder/decoder, a Conv1D decoder, and a Conv1D prediction layer. Filter sizes vary from 32 to 160, with kernel sizes of 5 and 9, strides of 2, and activation functions featuring (Leaky ReLU) and Sigmoid. Each layer's specific parameters are detailed for exact execution

| | Conv1D encoder | | | LSTM encoder | | LSTM decoder | | Conv1D decoder | | | Conv1D pred |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | L1 | L2 | L3 | L4 | L5 | L6 | L7 | L8 | L9 | L10 | L11 |
| Filter size | 32 | 64 | 128 | 128 | 64 | 64 | 128 | 128 | 64 | 32 | 160 |
| Kernel size | 5 | 5 | 5 | – | – | – | – | 5 | 5 | 5 | 9 |
| Stride | 2 | 5 | – | – | – | – | – | – | 5 | 2 | – |
| Activation function | Leaky ReLU | – | | | | | | | | Leaky ReLU | Sigmoid |

The stride sizes 4 and 2 are used to build smaller feature maps of longer sequences to reduce complexities. The activation function of (LeakyReLU) is used at the processing step while sigmoid at the prediction step. These techniques enhance the model's accuracy. To get higher-level features, three layers of 1-D (CNN) (Conv1D) are employed as an encoder. The number of filter sizes increases with each layer because deeper layers can extract more precise information. Each 1-D (CNN) layer has the following batch normalization layer to normalize the input by maintaining the mean close to 0 and standard deviation near 1. This regularization technique develops a stable and faster model

[40]. Thus, high-level features are extracted and encoded from the input array, whereas dimensionality reduction also occurs when input is transferred into the network's deeper layer. This encoded and regularized input is passed through the temporal encoder (LSTM). A two-layer (LSTM) that performs an element-wise multiplication operation on input-to-state and state-to-state transitions to have better temporal features. The (LSTM) encoder interprets the audio sequence to precise the acoustic material as cell state vectors. The encoder output is a constant-length vector that holds internal state vectors. These encoded (LSTM) are decoded through the RepeatVector (LSTM) layer. This layer acts as an adapter to integrate the encoder and decoder of (LSTM). The initial states of the decoder are the final states of the encoder. With the help of these initial states, the decoder transforms the learned acoustic internal representation of the input audio sequence into the corresponding output sequence. These decoded (LSTM) layers assist in decoding the conv1D layer through Conv1DTranspose. This layer up-sampled the encoded data to larger data, i.e., mapping a 2 × 2 array to another 4 × 4 array. This layer performs both up sampling and convolution. It brings the encoded input sequence into its original shape [41]. Moreover, the final step is to predict the decoded information through Conv1D, which employs sigmoid as an activation function. It predicts a probability for each acoustic input sequence belonging to either of the two classes. If predicted input sequences are reconstructed, then the predicted probability is normal, but if input sequences are not reconstructed and represent high error, then the predicted probability is abnormal.

## *3.2 Visual Model*

### *3.2.1 Feature Extraction*

The visual dataset in this proposed model is made up of several recordings collected by (CCTV) cameras, each with a certain resolution that assures clarity. The procedure starts with frame extraction, which divides each movie into individual frames. This stage is critical for computer vision algorithms since each frame serves as a visual snapshot that captures key periods in the movie. These snapshots are critical to the anomaly detection model, which examines each frame to evaluate if the observed activity is normal or aberrant. The preciseness of this extraction process is crucial since it directly influences the accuracy of the model's predictions, ensuring that critical features are kept for successful analysis.

Data preparation is the next step after the frames have been extracted. This stage is vital since it improves the extracted frames, enhancing their quality, and identifying key aspects that impact the model's learning capacity. Preprocessing comprises scaling the frames to 240 × 240 pixels to ensure consistency throughout the collection. This standardization promotes uniformity and makes the data simpler for the model to process. Furthermore, the frames are transformed from (RGB) to grayscale, which reduces computing complexity by simplifying color information while preserving crucial features. This stage optimizes the frames so that the model can process them more effectively while still keeping the relevant visual information.

Finally, during the preprocessing stage, the pixel values are normalized from 0 to 255 to a scaled range of 0 to 1. This normalization optimizes the data for machine learning by stabilizing and optimizing the model's learning process. The generated data is arranged in a structured manner, commonly written as (video_files, frame_sequence, feature_length), where the parameters correspond to the number of videos, the sequence of frames, and the length of the extracted features. This rigorous preparation guarantees that the anomaly detection model receives high-quality input data, allowing it to properly distinguish between normal and abnormal occurrences in video sequences.

### *3.2.2 Model Architecture*

Fig. 3 exhibits the visual model, while Table 3 shows the architecture of the model. The model is comprised of two sections: the first one is a spatial autoencoder, and the second one is a temporal autoencoder. The spatial features are related to the location of objects, while the temporal features are related to the motion of objects. The proposed model is composed of different Keras layers. A base model is developed with a sequential layer as input, which is a frame sequence. The model contains 8 layers: 1 input layer, 6 hidden layers, and 1 output layer.

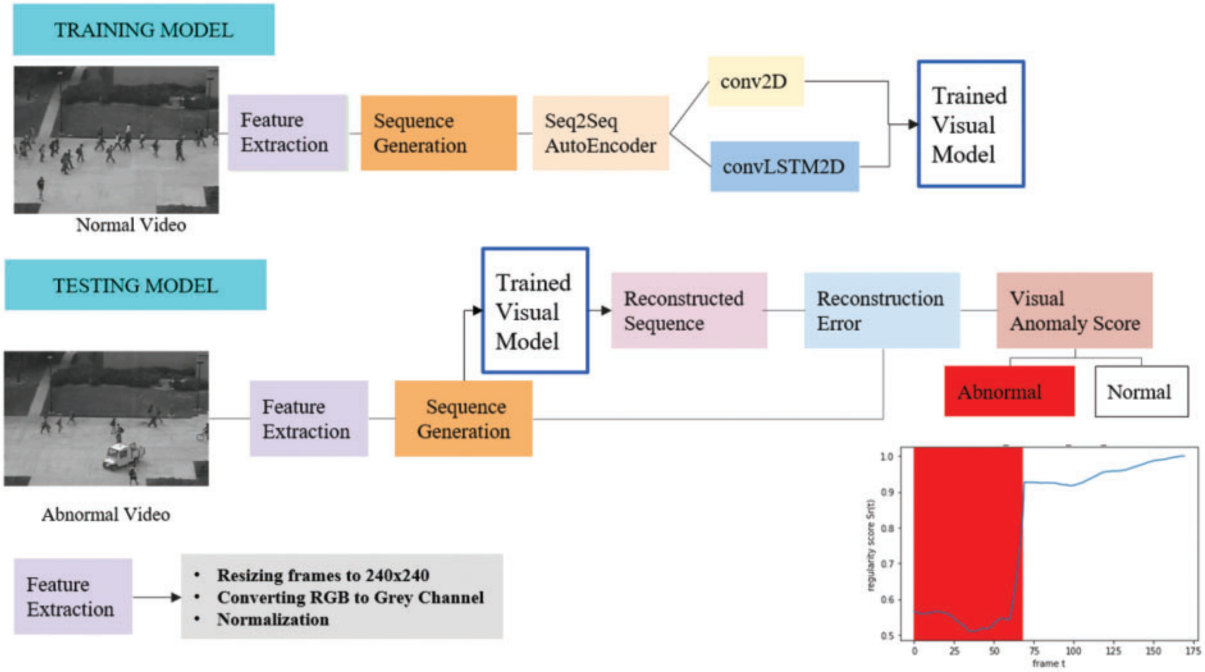


**Figure 3:** Block diagram: The video is preprocessed to extract explicit features of input data upon which the Seq2Seq autoencoder is trained. For testing, the trained model predicts the video frame sequences. The regularity score is calculated to determine anomalies through threshold value

**Table 3:** Architecture: The table illustrates the design of a neural network for (VAD) in video surveillance. It describes the layer configurations, including filter sizes, kernel sizes, strides, and activation functions for various network components such as the Conv2D encoder, ConvLSTM2D encoder decoder, Conv2D decoder, and Conv2D prediction layers

| | Conv2D encoder | | ConvLSTM 2D encoder decoder | | | Conv2D decoder | | Conv2D pred |
|---|---|---|---|---|---|---|---|---|
| | Layer1 | Layer2 | Layer3 | Layer4 | Layer5 | Layer6 | Layer7 | Layer8 |
| Filter size | 32 | 64 | 64 | 32 | 64 | 64 | 32 | 1 |
| Kernel size | (5, 5) | (5, 5) | (3, 3) | (3, 3) | (3, 3) | (5, 5) | (5, 5) | (7, 7) |
| Stride | (4, 4) | (2, 2) | | | | (2, 2) | (4, 4) | |
| Activation function | Leaky ReLU | – | – | – | – | – | Leaky ReLU | Sigmoid |

These layers are concatenated to build up a Seq2Seq AE. The model receives preprocessed input frame sequences of specific length and size. The model is designed with a particular number of filters and kernel size. The number of filters increased as the layers increased because early layers in the network learned few convolutional filters while layers deeper in the network learned more convolutional filters. The kernel size is set to (5 × 5) because there is a rule that if the input size is greater than (128 × 128) always use a kernel size greater than (5 × 5). The stride is used instead of max pooling because it reduces the dimension by keeping intact all the pixel information without discarding it; strides with smaller values capture fine details. To build up the spatial autoencoder, we have used three two-dimensional convolutional layers (Conv2D) to generate a computationally efficient autoencoder. For this autoencoder, the convolutional operation is preferred as it keeps intact all the spatial features among frames' pixels. It learns all of these features by convolutional, which is a dot product between input regions and filters. Temporal autoencoder is designed through three two-dimensional convolutional layers Conv2DLSTM. In this layer, element-wise multiplication operation is replaced by convolution, and weights are applied to input-to-state and state-to-state transitions to have better spatiotemporal features. The correlation between spatiotemporal features is built up through a convolutional layer. Moreover, (ConvLSTM) can predict the future state by considering all the current and past information of its neighbors. Batch normalization is employed between each layer of (AE) as it takes input from the previous layer, normalizes it, and passes this normalized output as input to the next layer. This maintains the efficient distribution of data by solving the internal covariate shift. This leads to faster convergence and the best accuracy scores. This layer is generally used as a regularization technique [40]. A Time Distributed Layer is applied to (AE) layers to maintain the relation between time-series input and its corresponding output independently for each time step. This layer is time-efficient and efficiently determines the peculiar features as it is applied to every temporal slice of the input. If this layer is not applied to sequential data, then the output of layers gets fused with each time step. This leads to unnecessary interference with different time steps and the inability to obtain separate time step values [42]. The (LeakyReLU) function is used as the activation function in both the convolutional and (ConvLSTM) layers, while a sigmoid function is applied at the prediction layer to predict the frame sequence with the probability of being either normal or anomalous. If frames are correctly reconstructed by (AE), then the anomaly score is high; otherwise, it is low.

### *3.2.3 Acoustic-Visual Model Inference*

The inference of the acoustic-visual model is performed individually. Initially, the features and preprocessing of both datasets are performed in the manner described in the preceding section. After that, the sequence generation technique is carried out. It is applied through the sliding window technique. It is a dynamic technique for extracting smaller sequences of a specific length from long sequences. The test data contain a distinct number of acoustic and visual frames. This technique is applied to these frames to get the continuous sequence of 10 frames. The ten frame sequences were selected with the concept that increasing the value of subsequent frames resulted in a better regularity score, but higher values of frame sequence slow the training process [43]. Following this, the trained model is tested upon these sequences. The model takes input sequences and reconstructs these input sequences. In this sequence reconstruction step, normal sequences are reconstructed accurately, while abnormal sequences are reconstructed imprecisely. Based on this accuracy and imprecision, reconstruction errors are calculated. This error is the Euclidean distance between input and reconstructed sequences. The accurately reconstructed sequence has a lower error, while the imprecisely reconstructed sequence has a high error. This error is normalized to acquire regularity score $s_r(t)$. In particular, each sequence's regularity score $s_r(t)$ is evaluated, where t is the number of frames, and $s_r(t)$ begins at frame (t) and concludes at frame $(t + 9)$ [44]. These $s_r(t)$ are also termed anomaly scores, where the high value of $s_r(t)$ depicts a normal frame and the low value illustrates an anomalous frame.

$$Anomaly\ Score = \begin{cases} s_r\ (t) > \gamma \text{ sequence is normal} \\ s_r\ (t) < \gamma \text{ sequence is abnormal} \end{cases} \tag{1}$$

### *3.3 Joint Model*

It is a supervised approach that is built upon the concept of late fusion that utilizes the outcome of each model by integrating them at the decision level. The proposed approach trains the two single modality costs; you can visualize the anomaly score of these models, which are concatenated to predict the outcome of sequences.

#### *3.3.1 Model Architecture*

The block diagram of the joint model is depicted in Fig. 4, while Table 4 shows the architecture of the model. Initially, a sequential model is built to maintain the acoustic-visual sequences. The model is comprised of 9 layers: 1 input layer, 7 hidden layers, and 1 output layer. These hidden layers are stacked in dense layers.

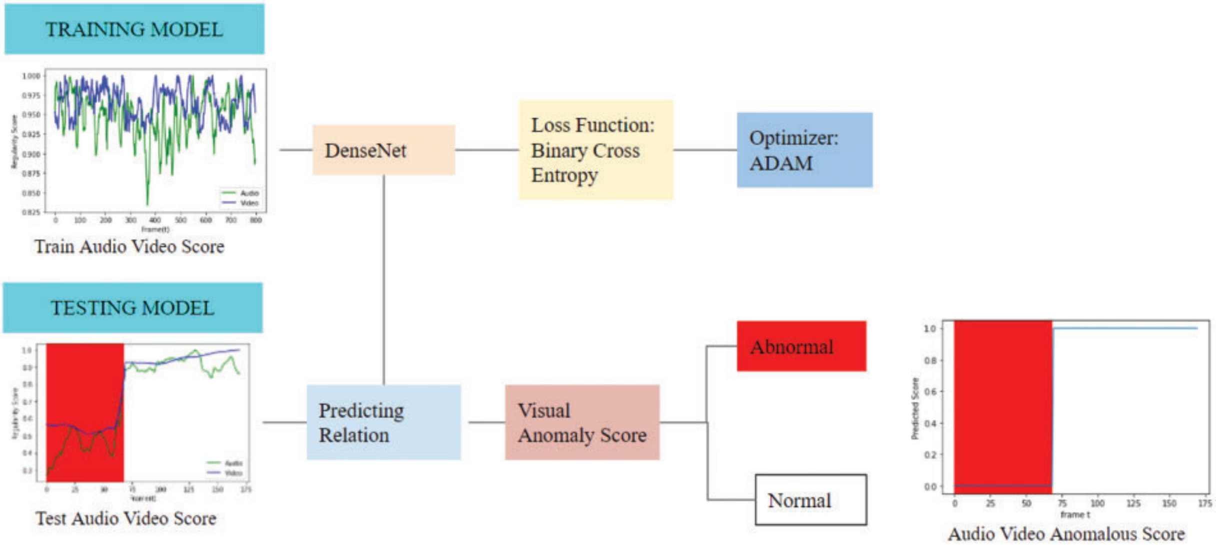


**Figure 4:** Block diagram: The anomaly scores of the acoustic and visual models are fused as input. This input trained the joint model through a loss function and an optimizer to generate the decisive results of the proposed model

The model built up with dense layers learns the association among scores and labeled data. With each dense layer, the unit size increases. The unit size defines the size of output from the dense layer. Thus, an increase in unit size causes a greater number of neurons in deeper layers. These layers learn the relationship between these two features: audio anomaly score and video anomaly score more accurately and precisely. These stacked dense layers deal with (LeakyReLU) as an activation layer to learn from input sequences. The dense layer of the joint model is trained to learn features from previous layers, while the last layer predicts the probability of each feature belonging to a specific class. For this purpose, Conv1D is applied to have an activation layer as a sigmoid to predict the class of each sequence, either normal or anomalous. The model is regularized through batch normalization to maintain the efficient distribution of data by solving the internal covariate shift. This leads to faster convergence and the best accuracy scores. With dense layers, the training process speeds up because it performs linear operations with every input processed by the function to generate output. It can learn the true relationship between features as it lets the neural network learn input related to the output.

**Table 4:** The architecture of the joint model: Each layer, along with (LeakyReLU), maintains steady gradients, numerical stability, increasing learning efficiency, improving feature combination, and overall network performance to predict the type of event, either normal or anomalous, using Sigmoid

| | Deep dense layer | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Layer1 | Layer2 | Layer3 | Layer4 | Layer5 | Layer6 | Layer7 | Layer8 | Layer9 |
| Units | 8 | 16 | 32 | 64 | 128 | 256 | 512 | 1024 | 1 |
| Activation function | Leaky ReLU | Leaky ReLU | Leaky ReLU | Leaky ReLU | Leaky ReLU | Leaky ReLU | Leaky ReLU | Leaky ReLU | Sigmoid |

### *3.3.2 Joint Model Inference*

The joint model is tested on the predicted joint acoustic visual anomaly scores. These scores act as an input for the model, and the input is already processed, so no preprocessing step is performed while their labeled output is considered as actual values. When these inputs are passed towards the trained model, it predicts the class of each input. The predicted class is either 0 or 1. The trained model predicts each acoustic visual frame as either normal or abnormal by plotting a graph between predicted scores and frames. The results of this model are decisive and dependent. Any (FP) in tested anomaly scores will affect its results, but due to training on the DenseNet, it learned the correlation among them. That's why it deals with this (FP) decisively.

$$Predicted\ Class = \begin{cases} 1 & class\ is\ normal \\ 0 & class\ is\ abnormal \end{cases} \tag{2}$$

## *3.4 Anomaly Detection*

### *3.4.1 Threshold Value ($\gamma$)*

$\gamma$ is a judgment value that predicts if the sequence is either normal or abnormal. If the predicted value exceeds $\gamma$, consider it normal or otherwise abnormal. The threshold value varies depending on the data type. The optimal $\gamma$ for the proposed model is based on high (TRP) as it defines accurate prediction of abnormal data. The modal is tested upon distinct $\gamma$, and those values are retained where the (TPR) value is high.

### *3.4.2 Regularity Score $s_r(t)$*

The lower the regularity score, the greater the chance of an anomalous event. To calculate this score, calculate the reconstruction error between the original frame and the predicted frame through Euclidean distance.

$$e(t) = ||x(t) - f_w(x(t))||_2 \tag{3}$$

Here, $e(t)$ is the reconstruction error, $x(t)$ is the original frame, and $f_w(x(t))$ is the predicted frame.

Now the regularity score is calculated based on $n\_e(t)$ as represented below:

$$s_r\,(t) = 1 - n\_e(t) \tag{4}$$

The predicted regularity score $s_r(t)$ determines whether each frame is either abnormal or normal through a defined threshold $\gamma$. A frame at t is marked as anomalous if $s_r(t)$ is less than $\gamma$. Contrary to this, a frame at t is marked as normal if $s_r(t)$ is greater than $\gamma$.

## 4 Experiment Results and Analysis

### *4.1 Datasets*

The proposed model is trained on a normal video dataset but tested on both normal and abnormal ones. The two different benchmark datasets are used to train and test the proposed model. The first one is the Anomaly Detection dataset from the Monitoring Human Activity dataset from the (UMN) dataset [45], the second one is (TUT) Rare Sound Events 2017 [46] and (UCSD PED 2) [47]. The statistics of these datasets are explained in Table 5.

**Table 5:** The table contrasts video and audio dataset statistics for (UCSD PED2) and (UMN), such as frame counts, anomaly details, and overall information. It identifies changes in total frames, training and test frames, normal and abnormal frames, and anomaly features between each dataset

| | UCSD PED 2 | | UMN | |
|---|---|---|---|---|
| | Audio | Video | Audio | Video |
| Frames information | | | | |
| Total frame | 4543 | 26876 | 7738 | 64686 |
| Training frame | 2533 | 24866 | – | 61446 |
| Test frame | 2010 | 2010 | – | 3240 |
| Normal frame | 2887 | 25220 | 6633 | 63581 |
| Abnormal frame | 1656 | 1656 | 1105 | 1105 |
| Anomalies information | | | | |
| No. of anomalies | 21 | 21 | 21 | 21 |
| No. of scenes | 1 | 1 | 3 | 3 |
| Anomaly type | 5 | 5 | 1 | 2 |
| General information | | | | |
| Resolution | 360 × 240 | | 320 × 240 | – |
| Ground truth | Spatial, Temporal | Temporal | Temporal | Temporal |
| Sampling rate | – | 22 kHz | – | 22 kHz |
| Open set | ✓ | ✓ | ✓ | ✓ |

### *4.1.1 Video Anomaly Dataset*

*UCSD Dataset:* The (UCSD PED 2) is comprised of 16 training videos and 12 testing videos acquired from a camera suspended above pedestrian pathways at distinct angles. The training and testing videos contain 119, 149, and 179 frames, while anomalies are carts, wheelchairs, skaters, and bikes.

*UMN Dataset:* The (UMN) dataset is comprised of three different environment scenarios: a plaza, interior, and lawn. The normal frames are random motions of the crowd, while abnormal frames are scattering of people in one direction from a central point. In the proposed method, the single video file is segmented into 6-s files to train and test each environment scenario separately.

### *4.1.2 Audio Anomaly Dataset*

*Audio UCSD PED2:* The normal acoustic data is attained through (TUT) Rare Sound Events 2017 [23], which is a 30 s file. The anomalous data is collected from several YouTube and Google websites and includes noises of carts, wheelchairs, skaters, and bikes. This data is comprised of 4.8, 6, and 7.2 s.

*Audio UMN Dataset:* The acoustic data of the file is prepared manually using (TUT) Rare Sound Events 2017 [23]. The anomaly in audio data is yelling from the crowd along with gunshot sounds. These audio files are composed in synchronization with (UMN) video files.

## **4.2 Model Parameters**

The loss function and optimizer are two model parameters that significantly improve the accuracy of the model and minimize the reconstruction error for every normal video. At each epoch, the optimizer evaluates and updates weights and biases, while the loss function minimizes the errors through updated weight and bias.

### *4.2.1 Losses*

(MSE) and (BCE) are applied as loss functions in the proposed model. MSE is best suited for reconstructing sequences [48] in (AE) while (BCE) is preferred for the joint model as it determines how much predicted values deviate from an actual value. The lower the loss, the more accurate the model is [49].

### *4.2.2 Optimizer*

(ADAM) is utilized as an optimizer to optimize parameters. It is an adaptive learning rate method, i.e., it evaluates individual learning rates for distinct parameters [50]. It is an optimization algorithm that requires less memory and faster running time [51].

### *4.2.3 Epochs*

It refers to one full iteration of the proposed model over the training dataset. Each training dataset has a different epoch defined in Table 6. The epochs of (UCSD PED 2) are user-defined, while the epochs of the (UMN) dataset are specified by the early stopping criteria. Batch normalization is added after each layer of (AE) as it regularizes output and speeds up the time and convergence.

**Table 6:** The table details the number of epochs and thresholds used to train audio and video models on several datasets, such as (UMN) Interior, Lawn, Plaza, and (UCSD PED 2)

| Dataset | Audio | | Video | |
|---|---|---|---|---|
| | Epoch | Threshold | Epoch | Threshold |
| UMN Interior | 35 | 0.85 | 31 | 0.85 |
| UMN Lawn | 41 | 0.85 | 44 | 0.85 |
| UMN Plaza | 38 | 0.8 | 40 | 0.7 |
| UCSD PED 2 | 50 | 0.85 | 300 | 0.92 |

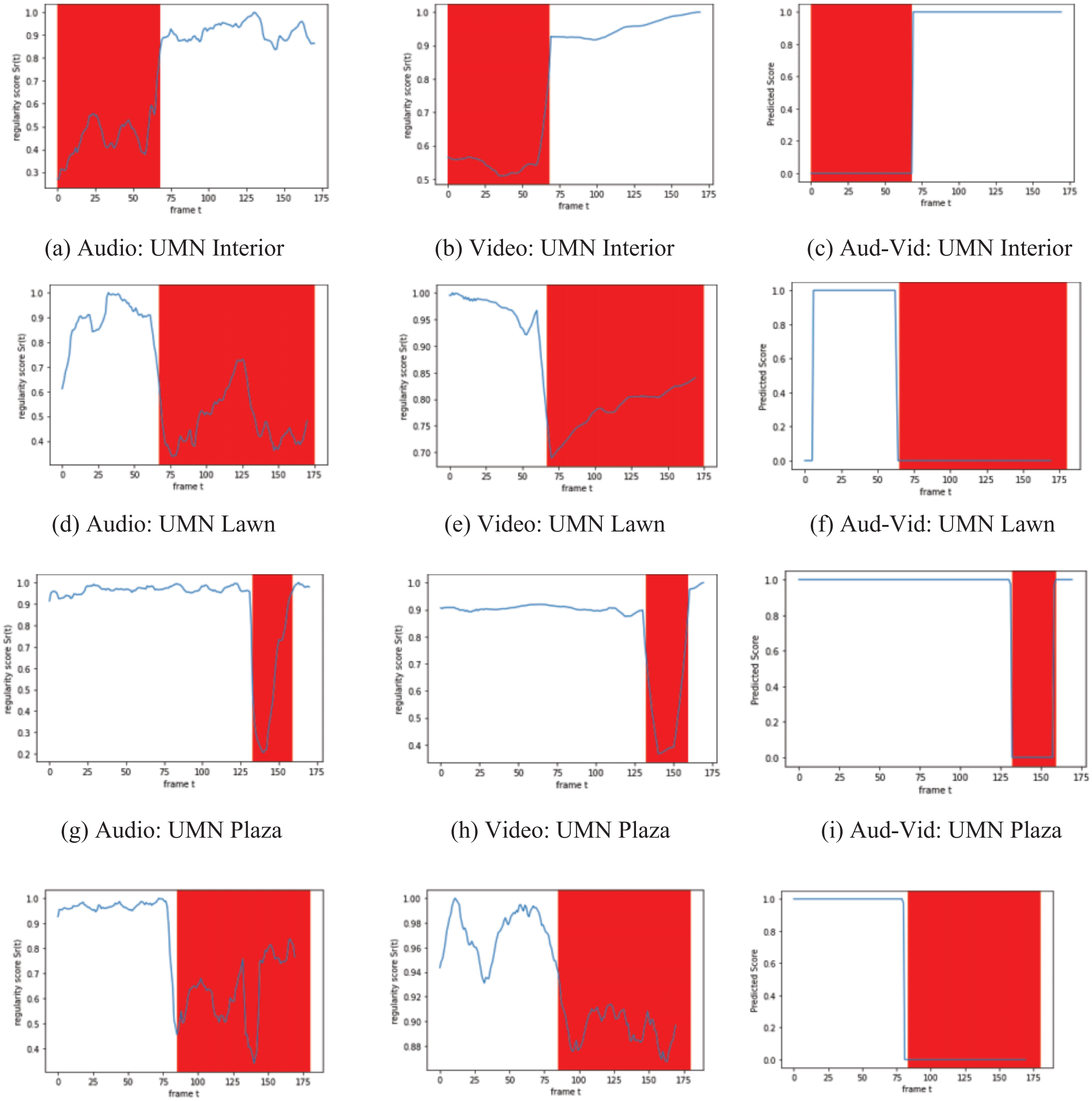
### *4.3 Implementation Details*

*Data Augmentation*

The data augmentation technique is employed upon the normal video dataset by generating n samples, and each sample contains f sequences/frames. This technique begins by generating n samples from the regular video dataset; each has f sequences or frames retrieved from the movie. The crucial part of this approach is how the frames are chosen and concatenated, using varying strides to introduce variety. In stride-1, all f frames in a sample are consecutive. For example, if the chosen frames may be frames 1, 2, 3, 4, and 5. This guarantees that the frames are in the same sequential order as they are presented in the original footage. Stride-2 skips one frame between each specified frame. For example, if $f = 5$, the frames may be 1, 3, 5, 7, and 9, resulting in a minor temporal gap between successive frames. Finally, in stride-3, two frames are skipped between each chosen frame, yielding a sequence such as frame 1, frame 4, frame 7, frame 10, and frame 13 if $f = 5$. This results in an even wider temporal gap between frames, broadening the samples. The goal of employing varied strides for frame selection is to increase variety in the training data. without requiring additional video sources. By changing the temporal connections between frames, the model is exposed to new patterns and variances in the typical video data. This increased variety improves the model's generalization ability to detect abnormalities. During training, the model develops to recognize regular patterns across a wider range of sequences, which might help it detect anomalies from these patterns in real-world circumstances. In conclusion, our data augmentation strategy fills the training dataset with diverse sequences, resulting in a more robust model. By exposing the model to diverse temporal patterns using stride-1, stride-2, and stride-3, the approach increases the model's generalization capabilities, resulting in more accurate anomaly identification in video surveillance. Hence, this is data augmentation in temporal dimension [52]. Our model is trained on a GPU, while the testing of the model can be performed on both GPU and CPU. The batch size is set equal to 10, along with 33% of the validation dataset.

### *4.4 Results*

The evaluation metrics employed for determining results are precision, recall, accuracy, and F1-score. Fig. 5 illustrates the results of the proposed model graphically, while Table 7 represents it quantitatively. In Fig. 5, the blue lines depict model prediction, while the region covered with red illustrates the ground truth for an anomalous area. These results have illustrated that integrated results are better than individual results because late fusion guarantees that numerous models are dealt with individually, including their defects and accuracy, resulting in an uncorrelated mistake that does not damage the joint model.



(a) Audio: UMN Interior

(b) Video: UMN Interior

(c) Aud-Vid: UMN Interior

(d) Audio: UMN Lawn

(e) Video: UMN Lawn

(f) Aud-Vid: UMN Lawn

(g) Audio: UMN Plaza

(h) Video: UMN Plaza

(i) Aud-Vid: UMN Plaza

(j) Audio: UCSD PED2

(k) Video: UCSD PED2

(l) Aud-Vid: UCSD PED2

**Figure 5:** The anomaly scores of the acoustic, visual, and joint model upon the UMN interior (a–c), UMN Lawn (d–f), UMN Plaza (g–i), and UCSD PED2 (j–l) dataset

**Table 7:** The table displays the proposed model's results via four separate metrics: precision, recall, accuracy, and F1-scores. All of these metrics are based on the (UMN) and (USCD PED 2) audio, video, and audio-video datasets

| Dataset | Precision | Recall | Accuracy | F1-score |
|---|---|---|---|---|
| UMN Interior | | | | |
| Audio | 0.9356 | 0.9176 | 0.9325 | 0.9265 |
| Video | 0.8321 | 0.9901 | 0.9287 | 0.9072 |
| Audio-Video | 0.9521 | 0.9297 | 0.9537 | 0.9408 |
| UMN Lawn | | | | |
| Audio | 0.9079 | 0.9512 | 0.9124 | 0.9290 |
| Video | 0.6503 | 1 | 0.8058 | 0.7881 |
| Audio-Video | 0.8886 | 0.9763 | 0.9427 | 0.9304 |
| UMN Plaza | | | | |
| Audio | 0.7741 | 0.9128 | 0.9659 | 0.8377 |
| Video | 0.7096 | 1 | 0.9513 | 0.8301 |
| Audio-Video | 0.8579 | 0.9082 | 0.9599 | 0.8823 |
| UCSD PED 2 | | | | |
| Audio | 0.9436 | 0.9502 | 0.9128 | 0.9469 |
| Video | 0.9061 | 0.8128 | 0.7861 | 0.8569 |
| Audio-Video | 0.9666 | 0.9546 | 0.9259 | 0.9605 |

*State-of-the-Art Methods*

The performance of our joint model is compared with other state-of-the-art methods. The evaluation metrics employed are (AUC) and (EER). The (AUC) calculates the overall performance of the binary classifier by plotting the (TPR) which is the rate at which the model accurately identifies actual positives, against the (FPR) which is the rate at which the model incorrectly classifies actual negatives as positives. Higher (AUC) values specify better model performance. The (EER) is the point on a receiver operating characteristic curve where the false positive and false negative rates are equal. It denotes a point when the rates of improper acceptances and rejections are equal. A lower (EER) implies improved model performance. The performance comparison is shown in Table 8. Our model demonstrates that the availability of both audio and video in the dataset improves the precision and accuracy of the model. The statistical analysis of the results is described below. In the case of (UMN) Interior, the precision, recall, accuracy, and F1-score of audio-video are better than the rest of the individual models. For (UMN) Lawn, accuracy and the F1-score of the joint model have shown better results than the rest of the singular modalities. The reason is the low precision rate of video for predicting the anomalies; its poor regularity score has dampened the precision and recall of the joint model as compared to the audio model. For (UMN) Plaza, the precision and F1-score of the audio-video model have performed well. The good point is that the precision score of audio and video separately is low, but it enhances the precision of the score of the joint model, which is the main motivation of this proposed approach. For (UCSD PED 2), all the evaluation metrics performed well for the audio-video data.

**Table 8:** The table displays the (AUC) and (EER) metrics, with higher (AUC) and lower (EER) values indicating better performance. The proposed method achieves the highest (AUC) and the lowest (EER) on both datasets, demonstrating improved performance

| Methods | AUC (%) | EER (%) |
|---|---|---|
| UCSD PED 2 | | |
| Pang et al. [16] | 83.2 | – |
| Feng et al. [53] | 83.8 | – |
| Feng et al. [53] | 84.5 | – |
| Rashmiranjan et al. [52] | 88.3 | 11.3 |
| Tian et al. [54] | 89.6 | 15.9 |
| Tian et al. [54] | 90.2 | 20.3 |
| Proposed | **93.1** | **8.1** |
| UMN dataset | | |
| Buckchash et al. [55] | 82 | – |
| Leyva et al. [56] | 88.3 | 19.8 |
| Rehman et al. [29] | 90 | – |
| Sabih et al. [57] | 92.3 | – |
| Parate et al. [58] | 93.6 | – |
| Proposed | **94.9** | **5.9** |

For state-of-the-art methods, research has used convolutional autoencoders, distinct variations of (PCA), and intermediate fusion techniques for visual anomaly detection. Our proposed approach has shown that our joint model (AUC) and (EER) are better than the visual model. Our (EER) value is much lower as compared to other papers. In the case of the (UMN) dataset, research has employed the techniques of Gaussian mixture models, Markov chains, bag-of-words, (SVM), and Grassmann manifolds. The (AUC) of these techniques is lower than our proposed approach because they have used traditional machine learning approaches whose results are not as accurate and robust as compared to deep learning techniques.

## 5 Conclusion

In this paper, a joint Seq2Seq (AE) and DenseNet model is proposed that utilizes acoustic and visual datasets to predict integrated anomaly scores. These datasets are trained independently upon respective acoustic and visual models. These Seq2Seq (AE) models processed the input sequence and generated the reconstructed output sequences to calculate anomaly scores. If input and output sequences are non-equivalent, the reconstructed error increases while the anomaly score reduces, indicating the sequences as abnormal and vice versa. Thus, the anomaly score of these models acts as an input to the DenseNet joint model; it trains itself upon these inputs and predicts the class either 0: anomalous or 1: normal. The acoustic and visual scores are independent, while the joint score is dependent upon them. The performance of the proposed model is assessed on three standard benchmarks: (UCSD PED 2), (UMN) datasets, and (TUT) rare sound events. Accuracy, precision, recall, and F1-score are classification metrics employed to compute the performance of the model. It indicates that the model outperforms the joint scores instead of the individual scores, depicting the competitive advantage of the proposed multi-modal approach. The proposed technique is compared with state-of-the-art techniques that show low (EER) and high (AUC). (AUC) of 93.1 and (EER) of 8.1 are obtained by the model on the (UCSD) dataset, whilst (AUC) of 94.9 and (EER) of 5.9 are obtained on the (UMN) dataset.

**Acknowledgement:** A special thanks to all the authors whose profitable and successive work is highly acknowledged in defining, planning, and executing this research work. I sincerely acknowledge all my friends for their affection, goodwill, and moral support. This work was supported and funded by the Deanship of Scientific Research at Imam Mohammad Ibn Saud Islamic University.

**Funding Statement:** This work was supported and funded by the Deanship of Scientific Research at Imam Mohammad Ibn Saud Islamic University (IMSIU) (grant number IMSIU-RG23148).

**Author Contributions:** Sameema Tariq, Ata-Ur-Rehman and Maria Abubakar contributed in conceptualization, methodology, formal analysis, writing—original draft. Waseem Iqbal, Hatoon S. Alsagri, Yousef A. Alduraywish and Haya Abdullah A. Alhakbani helped in methodology, formal analysis, writing—review & editing, supervision the manuscript. All authors reviewed the results and approved the final version of the manuscript.

**Availability of Data and Materials:** All data sets mentioned in Section 4 are publicly available, assuring replicability and availability for future research. References [45–47] and [59] include detailed information and links to these datasets.

**Ethics Approval:** Not applicable.

**Conflicts of Interest:** It is stated that the research work presented in this manuscript consists of our own ideas and research work. The contributions and ideas from others have been duly acknowledged and cited in the dissertation.

## References

[1] H. Lin, J. D. Deng, B. J. Woodford, and A. Shahi, “Online weighted clustering for real-time abnormal event detection in video surveillance,” presented at the ACM Int. Conf. Multimed., Amsterdam, Netherlands, Oct. 2016, pp. 536–540.

[2] Y. Zhang, H. Lu, L. Zhang, and X. Ruan, “Combining motion and appearance cues for anomaly detection,” *Pattern Recognit.*, vol. 51, pp. 443–452, Jan. 2016. doi: 10.1016/j.patcog.2015.09.005.

[3] C. Stauffer, *Automated Audio-Visual Activity Analysis*. Cambridge, MA, USA, Rep: MIT-CSAIL-TR-2005-057, 2005.

[4] P. J. Pereira *et al.*, “Using deep autoencoders for in-vehicle audio anomaly detection,” *Procedia Comput. Sci.*, vol. 192, pp. 298–307, Jan. 2021.

[5] M. Pooyan, X. Zhang, M. Hamidi, and J. Zhang, “Deep learning-based anomaly detection for compressors using audio data,” presented at the Annu. Reliabil. Maintainability Symp. (RAMS), Orlando, FL, USA, Jan. 2021, pp. 1–7.

[6] E. Marchi, F. Vesperini, F. Eyben, S. Squartini, and B. Schuller, "A novel approach for automatic acoustic novelty detection using a denoising autoencoder with bidirectional LSTM neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Apr. 2015, pp. 1996–2000.

[7] Towards Data Science, "Denoising Autoencoders (DAE): How to use neural networks to cleanup your data," Towards Data Science. Accessed: May 20, 2022. [Online]. Available: https://towardsdatascience.com/denoising-autoencoders-dae-how-to-use-neural-networks-to-cleanup-your-data-cd9c19bc6915

[8] Towards Data Science, "Sparse autoencoder neural networks: How to utilize sparsity for robust information encoding," Towards Data Science. Accessed: May 20, 2022. [Online]. Available: https://towardsdatascience.com/sparse-autoencoder-neural-networks-how-to-utilise-sparsity-for-robust-information-encoding-6aa9ff542bc9

[9] Keras, "Building Autoencoders in Keras," Keras. Accessed: May 20, 2022. [Online]. Available: https://blog.keras.io/building-autoencoders-in-keras.html

[10] Keras, "Variational Autoencoders (VAE)," Keras. Accessed: May 20, 2022. [Online]. Available: https://keras.io/examples/generative/vae/

[11] P. Kumari and M. Saini, "Anomaly detection in audio with concept drift using adaptive Huffman coding," 2021, *arXiv:2102.10515*.

[12] F. Amir and T. A. Gulliver, "Unsupervised log message anomaly detection," *ICT Express*, vol. 6, no. 3, pp. 229–237, Sep. 2020. doi: 10.1016/j.icte.2020.06.003.

[13] Z. H. Janjua, M. Vecchio, M. Antonini, and F. Antonelli, "IRESE: An intelligent rare-event detection system using unsupervised learning on the IoT edge," *Eng Appl. Artif. Intell.*, vol. 84, pp. 41–50, Jan. 2019. doi: 10.1016/j.engappai.2019.05.011.

[14] Z. Mnasri, S. Rovetta, and F. Masulli, "Anomalous sound event detection: A survey of machine learning based methods and applications," *Multimed. Tools Appl.*, vol. 81, no. 4, pp. 5537–5586, Feb. 2022. doi: 10.1007/s11042-021-11817-9.

[15] Y. Koizumi *et al.*, "Description and discussion on DCASE2020 challenge task2: Unsupervised anomalous sound detection for machine condition monitoring," 2020, *arXiv:2006.05822*.

[16] G. Pang, Y. Cheng, C. Shen, A. Van den Hengel, and X. Bai, "Self-trained deep ordinal regression for end-to-end video anomaly detection," presented at the IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Seattle, WA, USA, Jun. 2020, pp. 12173–12182.

[17] P. Kumari and M. Saini, "An adaptive framework for anomaly detection in time-series audio-visual data," *IEEE Access*, vol. 10, pp. 36188–36199, Jan. 2022. doi: 10.1109/ACCESS.2022.3164439.

[18] R. Morais, V. Le, T. Tran, B. Saha, M. Mansour and S. Venkatesh, "Learning regularity in skeleton trajectories for anomaly detection in videos," presented at the IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Long Beach, CA, USA, Jun. 2019, pp. 11996–12004.

[19] H. -S. Fang, S. Xie, Y. -W. Tai, and C. Lu, "RMPE: Regional multiperson pose estimation," presented at the IEEE Int. Conf. Comput. Vis. (ICCV), Venice, Italy, Oct. 2017, pp. 2334–2343.

[20] K. Cho *et al.*, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proc. Conf. Empir. Methods in Nat. Lang. Process. (EMNLP)*, 2014, pp. 1724–1734.

[21] B. Ramachandra, M. J. Jones, and R. R. Vatsavai, "A survey of single-scene video anomaly detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 5, pp. 2293–2312, May 2020. doi: 10.1109/TPAMI.2020.3040591.

[22] W. Chu, H. Xue, C. Yao, and D. Cai, "Sparse coding guided spatiotemporal feature learning for abnormal event detection in large videos," *IEEE Trans. Multimed.*, vol. 20, no. 1, pp. 246–255, Jan. 2018.

[23] Y. Cong, J. Yuan, and J. Liu, "Sparse reconstruction cost for abnormal event detection," presented at the IEEE Conf.Comput. Vis. Pattern Recognit. (CVPR), Colorado Springs, CO, USA, Jun. 2011, pp. 3449–3456.

[24] C. Lu, J. Shi, and J. Jia, "Abnormal event detection at 150 fps in MATLAB," presented at the IEEE Int. Conf. Comput. Vis. (ICCV), Sydney, Australia, Dec. 2013, pp. 2720–2727.

[25] Y. Lu, C. Cao, and Y. Zhang, "Learnable locality-sensitive hashing for video anomaly detection," 2021, *arXiv:2111.07839*.

[26] Q. Sun, H. Liu, and T. Harada, "Online growing neural gas for anomaly detection in changing surveillance scenes," *Pattern Recognit.*, vol. 61, pp. 187–201, Jan. 2017. doi: 10.1016/j.patcog.2016.09.016.

[27] L. Naud and A. Lavin, "Manifolds for unsupervised visual anomaly detection," 2020, *arXiv:2006.11364*.

[28] P. Wu *et al.*, "Not only look, but also listen: Learning multimodal violence detection under weak supervision," presented at the Eur. Conf. Comput. Vis. (ECCV), Glasgow, UK, Aug. 2020, pp. 322–339.

[29] A. -U. Rehman, H. S. Ullah, H. Farooq, M. S. Khan, T. Mahmood and H. O. A. Khan, "Multi-modal anomaly detection by using audio and visual cues," *IEEE Access*, vol. 9, pp. 30587–30603, Jan. 2021. doi: 10.1109/ACCESS.2021.3059519.

[30] D. Ramachandram and G. W. Taylor, "Deep multimodal learning: A survey on recent advances and trends," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 96–108, Nov. 2017. doi: 10.1109/MSP.2017.2738401.

[31] R. Gao, T. -H. Oh, K. Grauman, and L. Torresani, "Listen to look: Action recognition by previewing audio," presented at the IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Seattle, WA, USA, Jun. 2020, pp. 10457–10467.

[32] A. R. Lejbølle, B. Krogh, K. Nasrollahi, and T. B. Moeslund, "Attention in multimodal neural networks for person re-identification," presented at the IEEE Conf. Comput. Vis. Pattern Recognit. Workshops, Salt Lake City, UT, USA, Jun. 2018, pp. 179–187.

[33] X. Liu, L. Jiao, L. Li, X. Tang, and Y. Guo, "Deep multi-level fusion network for multi-source image pixel-wise classification," *Knowl.-Based Syst.*, vol. 233, Jan. 2021, Art. no. 106921. doi: 10.1016/j.knosys.2021.106921.

[34] X. Liu, L. Jiao, J. Zhao, J. Zhao, and D. Zhang, "Deep multiple instance learning-based spatial-spectral classification for PAN and MS imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 9, pp. 461–473, Sep. 2017.

[35] P. K. Atrey, M. A. Hossain, A. El Saddik, and M. S. Kankanhalli, "Multimodal fusion for multimedia analysis: A survey," *Multimed. Syst.*, vol. 16, pp. 345–379, Jan. 2010. doi: 10.1007/s00530-010-0182-0.

[36] K. M. Imran, S. N. Foley, and B. O'Sullivan, "Database Intrusion Detection Systems (DIDs): Insider threat detection via behavioural-based anomaly detection systems—A brief survey of concepts and approaches," 2020, *arXiv:2011.02308*.

[37] S. Dubnov, "Generalization of spectral flatness measure for non-gaussian linear processes," *IEEE Signal Process. Lett.*, vol. 11, pp. 698–701, Dec. 2004. doi: 10.1109/LSP.2004.831663.

[38] G. Sharma, K. Umapathy, and S. Krishnan, "Trends in audio signal feature extraction methods," *Appl. Acoust.*, vol. 158, Jan. 2020, Art. no. 107020. doi: 10.1016/j.apacoust.2019.107020.

[39] Analytics India Magazine, "A tutorial on spectral feature extraction for audio analytics," Analytics India Magazine. Accessed: Dec. 20, 2021. [Online]. Available: https://analyticsindiamag.com/a-tutorial-on-spectral-feature-extraction-for-audio-analytics/

[40] Analytics India Magazine, "Hands-on guide to implement batch normalization in deep learning models," Accessed: Jun. 13, 2021. [Online]. Available: https://analyticsindiamag.com/hands-on-guide-to-implement-batch-normalization-in-deep-learning-models/

[41] Analytics India Magazine, "Convolutional layer," Accessed: Jun. 12, 2021. [Online]. Available: https://databricks.com/glossary/convolutional-layer

[42] M. L. Value, "Time distributed layer in Keras with example in Python," Accessed: Jun. 13, 2021. [Online]. Available: https://valueml.com/time-distributed-layer-in-keras-with-example-in-python/

[43] M. Hasan, J. Choi, J. Neumann, A. K. Roy Chowdhury, and L. S. Davis, "Learning temporal regularity in video sequences," presented at the IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Las Vegas, NV, USA, Jun. 2016, pp. 733–742.

[44] Towards Data Science, "Prototyping an anomaly detection system for videos: Step-by-step using LSTM & convolutional," Towards Data Science. Accessed: Jun. 10, 2022. [Online]. Available: https://towardsdatascience.com/prototyping-an-anomaly-detection-system-for-videos-step-by-step-using-lstm-convolutional-4e06b7dcdd29

[45] Github, "UCSD-UMN anomalous audio datasets," Accessed: Aug. 05, 2022. [Online]. Available: https://github.com/tsameema/UCSD-UMN-Anomalous_Audio-Datasets

[46] DCASE, "Task 1: Rare sound event detection," Accessed: Jul. 24, 2021. [Online]. Available: https://dcase.community/challenge2017/taskrare-sound-event-detection

[47] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos, "Anomaly detection in crowded scenes," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, San Francisco, CA, USA, 2010, pp. 1975–1981.

[48] Machine Learning Mastery, "How to choose loss functions when training deep learning neural networks," Machine Learning Mastery. Accessed: Jan. 13, 2022. [Online]. Available: https://machinelearningmastery.com/how-to-choose-loss-functions-when-training-deep-learningneural-networks/

[49] Analytics Vidhya, "Binary cross-entropy (Log Loss) for binary classification," Analytics Vidhya. Accessed: Jan. 13, 2022. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/03/binary-cross-entropy-log-loss-for-binary-classification/

[50] Analytics Vidhya, "A comprehensive guide on deep learning optimizers," Analytics Vidhya. Accessed: Jan. 20, 2022. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/10/a-comprehensive-guide-on-deep-learning-optimizers/

[51] Towards Data Science, "Adam: Latest trends in deep learning optimization," Towards Data Science. Accessed: Jan. 20, 2022. [Online]. Available: https://towardsdatascience.com/adam-latest-trends-in-deep-learning-optimization-6be9a291375c

[52] N. Rashmiranjan, U. C. Pati, and S. K. Das, "Video anomaly detection using convolutional spatiotemporal autoencoder," presented at the Int. Conf. Contemp. Comput. Appl. (IC3A), Patna, India, Feb. 2020, pp. 175–180.

[53] J. Feng, Y. Liang, and L. Li, "Anomaly detection in videos using two-stream autoencoder with post hoc interpretability," *Comput. Intell. Neurosci.*, vol. 2021, no. 1, 2021, Art. no. 7367870. doi: 10.1155/2021/7367870.

[54] W. Tian, Z. Miao, Y. Chen, Y. Zhou, G. Shan and H. Snoussi, "AED-Net: An abnormal event detection network," presented at the Eng. Conf., Nanjing, China, Sep. 2019, pp. 930–939.

[55] H. Buckchash and B. Raman, "Towards zero shot learning of geometry of motion streams and its application to anomaly recognition," *Expert. Syst. Appl.*, vol. 186, Sep. 2021, Art. no. 114916. doi: 10.1016/j.eswa.2021.114916.

[56] R. Leyva, V. Sanchez, and C. -T. Li, "Video anomaly detection with compact feature sets for online performance," *IEEE Trans. Image Process*, vol. 26, no. 7, pp. 3463–3478, Jul. 2017. doi: 10.1109/TIP.2017.2695105.

[57] M. Sabih and D. K. Vishwakarma, "A novel framework for detection of motion and appearance-based anomaly using ensemble learning and LSTMs," *Expert Syst. Appl.*, vol. 192, Jan. 2022, Art. no. 116394.

[58] M. R. Parate, K. M. Bhurchandi, and A. G. Kothari, "Anomaly detection in residential video surveillance on edge devices in IoT framework," 2021, *arXiv:2107.04767*.

[59] Monitoring human activity, Accessed: Jul. 24, 2021. [Online]. Available: http://mha.cs.umn.edu/proj_events.shtml