Journal Européen des Systèmes Automatisés
Vol. 58, No. 4, April, 2025, pp. 669-687
Journal homepage: http://iieta.org/journals/jesa

X2_PDDVnet: An Explainable AI Based Dual Path Dense Dilated Vision Transformer Network Based Anomaly Detection

Shameem Akthar K.*, K. Lakshmi Priya

Department of Computer Science, Karpagam Academy of Higher Education, Coimbatore 641021, India

Corresponding Author Email: s.aktharu@gmail.com

https://doi.org/10.18280/jesa.580402

Received: 10 March 2025
Revised: 14 April 2025
Accepted: 21 April 2025
Available online: 30 April 2025

Keywords:
anomaly detection, Vision Transformer, explainable AI, zigzag feature learning, dilated multi-scale inception network and LIME

ABSTRACT

An essential function of video surveillance systems that are widely utilized for public safety and other purposes is automatic anomaly detection. The interpretability of anomaly detection is essential, though, because the kind and severity of the anomalies in the video dictate the appropriate response. Thus, the proposed study presents an explainable AI-based model, X2_PDDVnet, for video anomaly detection with high accuracy in the detection and interpretation of anomalous events. This model uses a two-path structure combining Vision Transformer (ViT) and AlexNet-inspired Convolutional Neural Network (CNN) hrough ZigZag path learning, which enhances feature extraction based on local and global patterns for such complex and dynamic video scenes. This hybrid approach advances video anomaly detection through combining global and local feature extraction to cover the analysis of high-level and granular details in any video frames. The three datasets include UCSD Anomaly Detection (University if Caifornia, San Diego), Avenue, and Shanghai Tech are utilized. Preprocessing of data is done by applying denoising, contrast enhancement, geometric transformations, and normalization to ensure optimized input. After the pre-processing, the model utilizes a dual-path encoder model. The ViT path captures global relationships between video frames by allowing each frame to be a sequence of tokens for the detection of spatial and temporal anomalies. In the meantime, the Zigzag Alex Net path makes use of dilated convolutions for local feature detection for further multi-scale information capture through an embedded Dilated Multi-Scale Inception Network (DiMS-Inception) in every convolution block. This dual-path structure yields a robust feature representation. The Grad-CAM and LIME offer heatmaps for improved visualization, which emphasize areas in crucial frames to make the mathematical model's decision-making process clearer. In this manner, the model succeeds in producing a transparent and interpretable anomaly detection process, which is of utmost importance for practical surveillance applications. The combination of ViT and CNN with the features of interpretability is promising for accurate and reliable anomaly detection in security systems.

1. INTRODUCTION

In the modern world, monitoring public violence is essential for the safety and safety of large cities. Consequently, video surveillance systems are now a crucial part of the growing Internet of Things (IoT)-based smart city projects. In many urban locations, video surveillance is required to observe anomalous activity like traffic accidents, robberies, and illicit activity. Computer vision researchers are very interested in methods that involve computer vision video monitoring because of the widespread use of surveillance cameras for public places [1]. The efficient use of surveillance applications in homes, workplaces, public areas, and enterprises makes abnormal event detection much more common. However, the irregularities cannot be detected in real-time by the traditional video processing approaches. Automating the detection of abnormal events is crucial in order to reduce time and expenses and take appropriate action before major problems arise [2]. An anomaly is an unexpected event that occurs in a crowded context; there may be more than one abnormality at the same time. Video anomaly detection (VAD) typically uses a temporal video segmentation algorithm to identify shot boundaries over successive video frames across multiple shot videos [3]. An effective and trustworthy anomaly detection technique is still required to manage the increasing amount of video data produced by these systems since video information is dynamic and complicated. Traditional approaches often rely on outdated automated systems or manual watching that is laborious, prone to errors, and ineffective. Compared to previous video analysis techniques, data imbalances among abnormal and normal segments contribute to it being more difficult to find and recognize abnormalities because abnormal actions are infrequent related to regular ones [4].

The ability to identify anomalous occurrences from a continuous video series is the foundation of video anomaly detection technology. Abnormal events, on the other hand, are ill-defined and boundaryless in the real world. A crucial task for a variety of video surveillance applications is identifying such occurrences. This motivates academics to develop an automated model for video data identification, segmentation, and classification [5, 6]. However, the use of hand-crafted features and heuristics in classic video anomaly detection algorithms limits their ability to handle a variety of complex circumstances. Artificial intelligence (AI)-based automatic anomaly detection solutions are necessary for surveillance systems due to the inefficiency of human observation [7]. Convolutional Neural Networks (CNN) are used by several video anomaly detection algorithms to learn temporal and spatial video data. The video is then rebuilt using inverse coding, or it can be combined with optical flow technology to forecast the subsequent frame. Usually, unsupervised, semi-supervised, or unsupervised learning is used in anomaly detection frameworks. A number of machine learning and computer vision methods have been put forth for the purpose of detecting anomalies in video sequences throughout the past ten years [8]. Significant advancements in VAD have been made likely by the emergence of deep learning. New methods that exceed earlier methods and solve the disadvantages of traditional machine learning (ML) have been made feasible by the usage of deep learning (DL) techniques. Additionally, even when an anomalous event occurs, its exact nature may not be fully known [9, 10].

A variety of DL models are employed for anomaly recognition and prevention, such as CNNs, long-short-term memory (LSTM), generative adversarial networks (GANs), recurrent neural networks (RNNs), and gated recurrent units (GRUs). But problems still exist because there aren't many anomalous actions, which means that datasets used for anomaly identification have considerably fewer positive samples (anomalies) than negative samples [11, 12]. The performance of the present methods is often worse due to a high percentage of false alarms. Despite their tremendous capacity, CNN-based methods are unable to detect long-term trends in the data due to the inherent localization of convolutional processes. The incapacity of CNNs to identify unanticipated irregularities that significantly diverge from the training data is one of its issues, as they are intended to recognize patterns in data [13]. A sequence-to-sequence model called Transformer has significantly improved video captioning, video prediction, and natural language processing (NLP). Transformers process the complete input data sequence in a single operation, as opposed to CNNs, which process the input sequences sequentially [14, 15]. In order to improve feature learning, Vison transformer-based anomaly detection is suggested in this study in terms of ZigZag path learning. The decision-making process involving video anomaly detection is made more transparent and comprehensible by integrating Grad CAM and LIME. The following are this work's primary contributions:

•The development of the X2_PDDVnet model, which combines a Vision Transformer (ViT) and AlexNet-based CNN through ZigZag path learning to improve video anomaly detection. This hybrid approach allows for the extraction of both global and local features, enhancing the model's capability to analyze complex video scenes effectively.

•The ZigZag Encoder and the Dilated Multi-Scale Inception Network (DiMS-Inception) within the model allow for comprehensive feature extraction at multiple scales.

•The Inception module's dilated convolutions enable the model to efficiently broaden its receptive area and catch long-range dependencies. This functionality is especially crucial for video analysis since abnormalities can show up in a variety of settings and forms.

•Additionally, the integration of explainable AI techniques, such as Grad-CAM and LIME, makes the model's decision-making process transparent involves creating heatmaps that show important regions that affect forecasts.

The structure of the paper is as follows: A synopsis of pertinent material is given in Section 2. The proposed strategy is examined in Section 3. Results are summarized in Section 4. Section 5 contains the paper's conclusion.

2. LITERATURE REVIEW

This section examines a few of the most recent research works regarding video and event anomaly detection using the deep learning techniques.

Kim et al. [16] have suggested a Cross U-Net framework that takes speed and accuracy of anomaly detection into account. The Cross U-Net architecture makes use of two U-Net-based subnetworks in a recently suggested deep learning model. In order to be utilized as the input for the next layer, it makes ensures that the result of each third layer in the contracting path merges with the output of the matching layer in the other subnetwork. The cascade sliding window approach, a recently suggested technique for calculating a frame's anomaly score, is also used in this framework. Ped2, Avenue, and Shanghai Tech datasets were used to evaluate the Cross U-Net framework's abnormality recognition accuracy.

Le et al. [17] developed a transformer-based TwinSformer model for anomaly detection in aerial drone-based videos, overcoming challenges such as dynamic backgrounds, changing viewpoints, and intricate textures. The model uses a multi-stage encoder to generate multi-scale features, which are then input into a hierarchical spatio-temporal transformer. Tested on both ground and aerial datasets, the model shows superior performance compared to current methods.

Balamurugan and Jayabharathy [18] have developed a model Bi-LSTM and hybrid CNN-based abnormal event detection models. This approach extracts spatiotemporal information from every single frame that was chosen from a sequence of frames using CNN with pre-trained models. The multi-layer Bi-LSTM receives these features and is able to accurately categorize the anomalous occurrences in complex road surveillance scenarios. To improve video summarization, the fine grain technique employs an LSTM encoder-decoder model based on hierarchical temporal attention.

Taghinezhad and Yazdi [19] introduced a novel anomaly detection technique based on unsupervised frame prediction that enhances overall performance. A Time-distributed 2D CNN-based decoder and encoder with a U-Net-like architecture was developed. The most pertinent archetypal pattern corresponding to the typical situation is retrieved and stored in memory slots by a memory module during training. This enables the model to produce inaccurate predictions in response to unusual input. Dilated convolutions were proposed as an upstream multi-branch structure to extricate contextual information while maintaining regular semantic patterns over several dimensions. The optical flow loss function is effectively replaced by the multi-path structure, which integrates time data into network architecture. The benchmark datasets CUHK Avenue, UCSD Ped1, and UCSD Ped2 are used to test the experiment.

Aslam and Kolekar [20] created a technique TransGANomaly, is a ground-breaking anomaly detection technique that uses a GAN based on a video Vision Transformer (ViViT). A video frame predictor that has only been adversarially trained on standard video data is the proposed framework. A ViViT network serves as the GAN's generator, receiving 3D input tokens from the video clips. The generator uses previous sequences to foresee the future frame. The original and expected frames for binary classification are then fed into the discriminator in the model. Several experiments have been carried out using the Shanghia Tech, CUHK Avenue, and UCSD Pedestrian datasets to verify the efficacy of the proposed method.

Wu et al. [21] presented an effective DL technique for detecting anomalies in movies by extracting high-level concept and context characteristics for training denoising autoencoders (DAE) from pre-trained deep models. The suggested approach requires little training time and delivers recognition results with the state-of-the-art techniques (less than 10 s on UCSD Pedestrian datasets). The combination of autoencoder as well as SHapley Additive exPlanations (SHAP) for model interpretability in video detection of anomaly is also used for the first time here.

Qasim and Verdu [22] have created an automated approach utilized a deep CNN and an SRU (Simple Recurrent Unit) to recognize abnormalities in videos. The SRU gathered temporal features, while the ResNet framework leveraged the incoming video frames to acquire high-level feature representations. Its highly parallelized implementation and expressive recurrence enhance the visual anomaly detection system's accuracy. ResNet18 + SRU, ResNet34 + SRU, and ResNet50 + SRU are the three models recommended in the study to identify abnormalities. The UCF-Crime dataset is used to analyze the suggested models.

Sharif et al. [23] have developed two pretrained feature extractors for ViT (such as CLIP) and CNN (such as C3D and I3D) to efficiently extract discriminating representations. Then, using the suggested temporal self-attention network (TSAN), video snippets of interest was presented while taking into account both short- and long-range temporal dependencies. CNN-ViT-TSAN is a generalized architecture based on multiple instance learning (MIL). It outlines a series of models for the WVAED difficulty using TSAN and characteristics that have been derived from CNN and/or ViT. Experiments conducted on well-known public crowd datasets proved that the CNN-ViT-TSAN model is effective.

Yang et al. [24] have presented a new two-stream fusion technique for identifying irregular events in order to better handle these various abnormal events. The object, posture, and optical flow features are initially extracted. The object and pose data are then combined early on to eliminate occluded pose graphs. A Spatio-Temporal Graph Convolutional Network (ST-GCN) is fed the trustworthy posture graphs in order to identify anomalous behavior. At the same time, a framework was presented for video prediction that uses the disparity amid expected and ground truth frames to detect anomalous frames. The final results are obtained by combining the prediction and classification streams at the decision-level.

Bajgoti et al. [25] have introduced Swin Anomaly, Swin Transformers are used for a conditional GAN-based autoencoder-based feature extractor technique for VAD. This suggested technique uses a 3D encoder to upsample spatiotemporal data from a series of video frames, and then a 2D decoder to predict a subsequent frame. Patch-wise mean squared error and Simple Online and Real-time Tracking (SORT) were used to track and identify anomalies in real time. The recommended strategy outperforms the current prediction-based video anomaly detection techniques and provides flexibility in identifying anomalies using a range of parameters.

Transformer-based models like ViViT, Swin Transformer, and CNN-ViT hybrids are used alone or not combined with multiscale local pattern detection processes. This work fills this gap using X2_PDDVnet, a dual-path architecture that combines ViT and a ZigZag-path augmented AlexNet for strong global-local anomaly detection. Compared to CNN-ViT-TSAN or SwinAnomaly, X2_PDDVnet's parallel encoder architecture retains hierarchical local details and frame-level global semantics, improving detection accuracy and explainability.

3. PROPOSED METHODOLOGY

The proposed methodology of the X2_PDDVnet model focuses on improving video anomaly detection by combining a ViT and an enhanced AlexNet-based CNN with ZigZag path learning. The ViT includes long-range dependencies by processing video frames as sequences of tokens, whereas the variant AlexNet focuses on local pattern recognition through the exploration of dilated convolutions and enlarging the receptive field by having Zigzag path learning. Finally, these two paths extract complementary global and local features and fuse together to form the comprehensive representation of the video input. To make the model more explainable, the integral of Grad-CAM with LIME enables users to visually inspect and understand the locations within video frames that are most relevant in driving anomaly detection decisions. The decoder of this model, resembling U-Net architecture, produces the segmentation maps from the fused feature maps before eventually using dilated convolutions to refine the output. This hybrid approach will detect anomalies at both the global and local levels in surveillance video, and the explainable AI components provide transparency into the process, making such a system reliable for security and monitoring systems. Figure 1 displays the suggested methodology's overall block diagram. The proposed methodology's block diagram is depicted in Figure 1.

3.1 Data collection

Initially, three open-source datasets are collected: the UCSD Anomaly Detection Dataset, which provides video sequences capturing normal and abnormal activities; the Avenue Dataset for Abnormal Event Detection, focusing on urban abnormal events; and the Shanghai Tech Dataset, which enriches the training data with diverse scenarios.

3.2 Preprocessing

Once collected, Preprocessing is done on the data, which includes geometric transformations like flipping it, rotation, cropping, and translation, along with noise injection to increase the number of training samples, denoising techniques like Gaussian filtering to eliminate noise, and histogram equalization to improve contrast and extract features. Further, pixel intensity values are normalized and standardized to enhance convergence of the model.

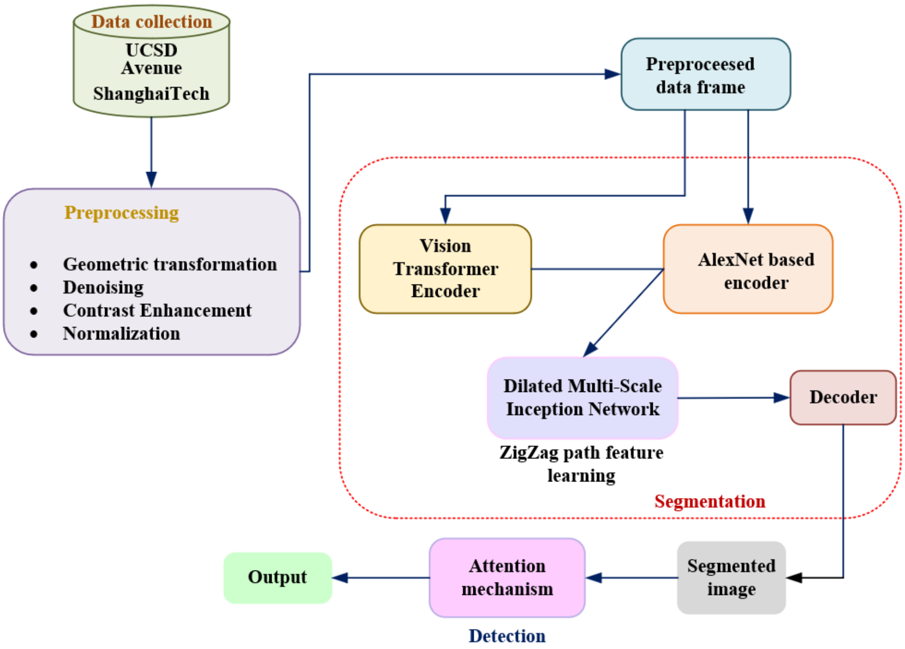


Figure 1. The proposed methodology's block diagram

3.2.1 Geometric transformation and data augmentation

A variety of transformations like cropping, rotation, flipping, translation, and noise injection are utilized to augment the dataset. These techniques help to create variations of the original images, providing the model with more diverse examples to learn from.

Flipping: Flipping the image can be done in an upward direction or on a level plane. It creates an image by rotating it by a factor of 90 degrees. Vertical flipping is not supported by all systems. The photo is first rotated 180 degrees to refine vertical flipping, and then level flipping is carried out.

Color space: Color space is transformed by photometric conversions. Using this technique, a three-photo stacked matrix is created, and each matrix has dimensions of height x width. The pixel values for every RGB color value are represented by this matrix. The lighting problems can be fixed by altering the image's color distributions.

Cropping: The process of enlarging a little segment of the original photograph to match its proportions is known as random cropping. If necessary, the image is resized using random cropping. Translation preserves the spatial dimensions of an image, as opposed to arbitrary cropping, which reduces their size.

Rotation: Depending on the requirements, the image can be rotated 90 degrees or located at small angles. An image rotated 90 degrees without any additional background noise once it has been aligned. On the other hand, this is not true when angled slightly. If the background is black or white, the newly introduced noise will merge in with the rest of the image. The network will recognize this as part of the image yet prevent it from fully blending in if the background of the image has distinct colours.

Translation: To find the object in any area of the picture, apply the translation idea. Positional bias in the data can be prevented by changing the image left, down, up, or right, or by moving it in the X or Y direction. It helps the network search the entire image, which causes background noise in the image.

Noise injection: Noise injection to enhance model generalization and avoid overfitting. Salt-and-pepper noise with densities ranging in training images introduces variations, maintains structural details, increases learning efficiency, and model robustness.

3.2.2 Gaussian filtering

Linear filters are a popular technique for eliminating noise when additive noise is present. They convolve the image using a constant matrix to provide a linear combination of neighborhood values. Filters based on Gaussian functions are particularly significant since their forms are simply characterized and both the inverse and forward Fourier transformations of a Gaussian function are both real Gaussian functions. Furthermore, a narrower frequency domain filter will attenuate low frequencies and increase smoothing/blurring, resulting in a broader spatial domain filter. In essence, these Gaussian filters which are frequently used for picture denoising are linear filters. Eq. (1) indicates that the further pixels are from the center of a Gaussian filter, the lower their weight becomes.

$$G_o(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \tag{1}$$

Gaussian filters suppress noise while maintaining picture features by averaging pixel values over a local region, assuming that pixels in a neighbourhood have close values and that images have uniform spatial variations.

3.2.3 Contrast enhancement

Histogram equalization is the technique of flattening out an image's grayscale degree value distribution. A cumulative distribution function, which is a computation of the histogram, is necessary to carry out histogram equalization. The following Eq. (2) is a definition of the cumulative distribution function.

$$f(k) = \frac{(N-1)}{m} \sum_{k=0}^{n} h(k) \tag{2}$$

$$n = 1,2,3,4 \dots .N - 1$$

where, *f(k)* is the cumulative distribution function for intensity level k. The total pixel value in a picture is denoted by m. The value of the gray pixel is N. A histogram of the gray value k is represented by $h(k)$. The frequency of gray levels utilized in an image can be determined using histogram equalization.

3.2.4 Normalization

Normalization facilitates improved model learning by ensuring that the features have comparable scales. Normalization is typically used in image processing to scale pixel intensity values to a particular range, like [0, 1] or [-1, 1]. Eq. (3) presents the normalization formula.

$$p = \frac{(x - x_{min})(max - min)}{(x_{max} - x_{min}) + min} \quad (3)$$

where, p is the converted input image, (min, max) is the input variable's specified range, and (x_{min}, x_{max}) is the minimum and maximum intensity values in the image and x is the original. Through these preprocessing steps, the collected datasets are transformed into a format that enhances the learning capabilities of the X2_PDDVnet model. Each technique contributes to improving data quality and diversity, ultimately leading to better anomaly detection performance. Figure 2 has been augmented to provide a more detailed and accurate representation.

3.3 X2_PDDVnet model for segmentation

The core of the proposed X2_PDDVnet model is its two-path structure, which includes a ViT in combination with a ZigZag path AlexNet based CNN. Each of the paths processes the same input but in different ways as it operates as an encoder so that the features are orthogonal to each other. The fusion of the two paths takes place at a later stage to enhance the representations of the features even further. The architecture revolves around two components, the ZigZag Encoder and the ViT Encoder. Both the encoders take the same input but treat it in a different way to extract complementary features. Once these features pass through each of the two encoders, they are combined for further processing by the Decoder to finalize segmentation and classification results. This compilation of approaches catalyzes the capturing of local as well as global contextual information within the model. Hence, the ZigZag Encoder offers local detail and spatial patterns in the input and the ViT Encoder is more capable of global relationships across the input frames. The Dilated Multi-Scale Inception Network (DiMS-Inception) presented at the individual block of the Zigzag Encoder helps capture multi-scale features and expand its receptive field. The proposed X2_PDDVnet architecture is shown in Figure 3.

The model input is an image or a video frame that is processed in parallel both by the ZigZag Encoder and the ViT Encoder. Since features produced by the two are different, there's a chance that the model could extract a wide range of information from an input.

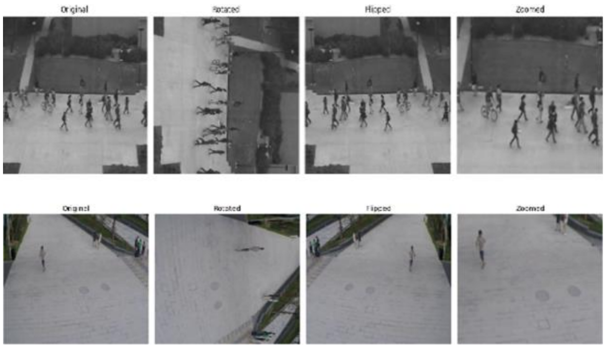


Figure 2. Augmented image

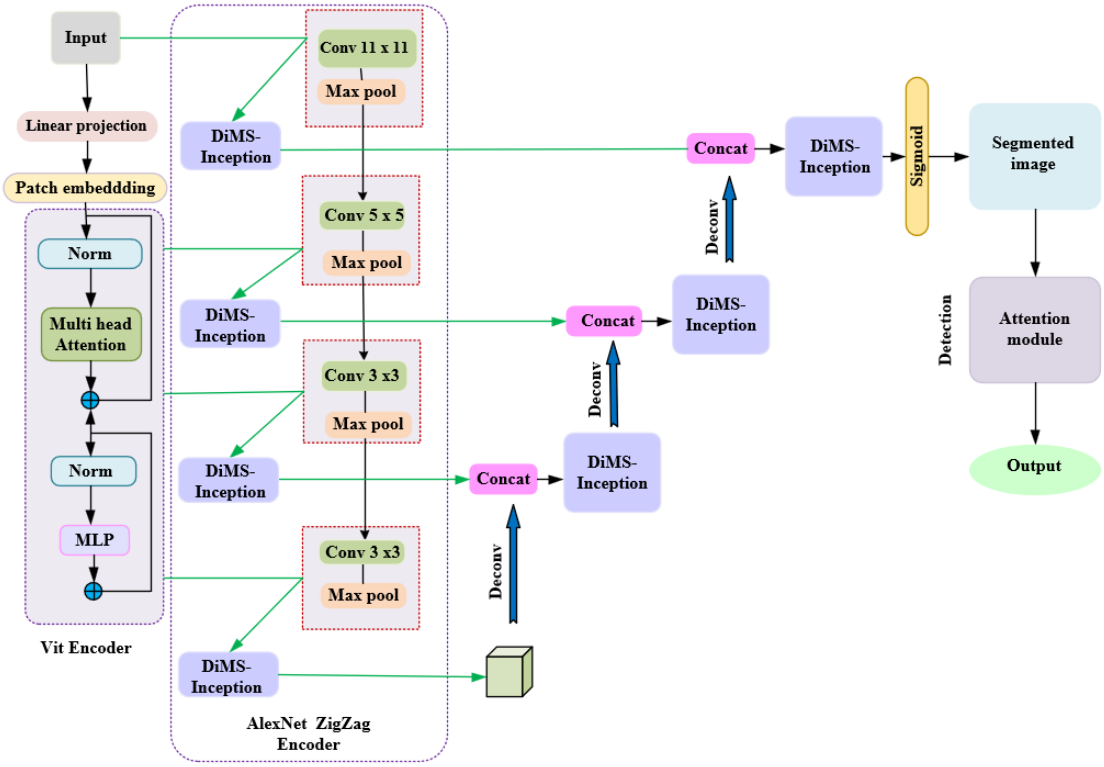


Figure 3. X2_PDDVnet architecture

ViTs have gained much popularity lately for their applicability to global relationships in image data. While CNNs are set to spend more time on local feature extraction by exploiting convolutional filters, a ViT takes an image as a sequence of patches and applies transformer layers to learn the relationship across the whole image. A one-dimensional sequence of token embeddings is the input for ViTs. To create a series of flattened tokens $x_p \in \mathbb{R}^{N\times(P^3.C)}$, an input image must divide into $x \in \mathbb{R}^{H\times W\times D\times C}$ with height H, width W, and D depth of the feature maps and C is the input channels. The flattened tokens in which (P, P, P) denotes each patch resolution and the length of the sequence is $N = \frac{(H\times W\times D)}{P^3}$. After that, a linear layer that remains constant across all of the transformer layers projects the patches upon a K-dimensional embedding space. To preserve the collected patches' spatial information, the projected patch embedding $\boldsymbol{E} \in \mathbb{R}^{(P^3.C)\times K}$ is added with a 1D learnable positional embedding in accordance with $\boldsymbol{E}_{pos} \in \mathbb{R}^{N\times K}$. The Transformer encoder's final input sequence, z_0 is:

$$z_0 = [x_v^1\boldsymbol{E}; x_v^1\boldsymbol{E}; \ldots\ldots, x_v^1\boldsymbol{E}] + \boldsymbol{E}_{pos} \tag{4}$$

A learnable class token, x_{class} is incorporated to the token sequence so that it can be used for classification tasks. Multiheaded self-attention (MHA) and multilayer perceptron's (MLP) are arranged in alternating layers within a single Transformer block to construct the Transformer encoder. Multiple stacked Transformer blocks make up a ViT, and the latent tokens $\boldsymbol{zi}$ are determined by

$$z_i{}' = MHA\big(Norm(z_{i-1})\big) + z_{i-1}, i = 1, \ldots.L \tag{5}$$

$$z_i = MLP\big(Norm(z_i{}')\big) + z_i{}', \qquad i = 1, \ldots.L \tag{6}$$

where, L -number of transformer layers, i - intermediate block identifier, and $Norm$ stands for layer normalization. MLP is made up of two linear layers with GELU activation functions. An MHA sublayer is composed of n parallel SA (self-attention) heads. The SA block is a parameterized function that specifically establishes the relationship between a query (q) and the matching representations of a key (k) and value (v) in a sequence $z \in \mathbb{R}^{N\times K}$. Eq. (7) is used to determine the attention weights (Attn) by measuring the similarity between two elements in z and their key-value pairs.

$$Attn = Softmax\left(\frac{qk^T}{\sqrt{K_h}}\right) \tag{7}$$

where, $K_h = K/n$ is a scaling factor, that keeps the number of parameters constant under various key k values. Eq. (8) uses the determined attention weights to calculate the output of SA for values v in the sequence z.

$$SA(z) = Attn.\boldsymbol{v} \tag{8}$$

where, $\boldsymbol{v}$ stands for the values in the input sequence. Furthermore, Eq. (9), which describes the output of MSA, in Eq. (9),

$$MHA(z) = [SA_1(z); SA_1(z); \ldots.; SA_n(z)]W_{mha} \tag{9}$$

where, the multi-headed trainable parameter weights are represented by $W_{mha} \in \mathbb{R}^{n\,K_h\times K}$. Using a ViT Encoder, the model captures both the local details as well as global contexts that capture anomalies that could be situated anywhere across any parts of an image or video frame. The Zigzag based feature learning encoder is depicted in the Figure 4.

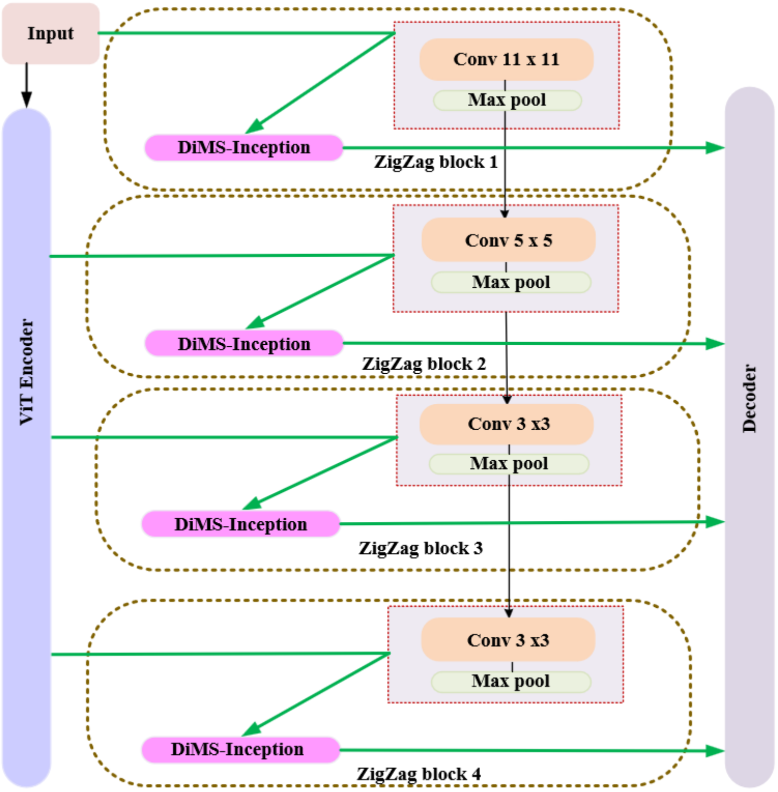


Figure 4. The architecture of Zigzag based feature learning encoder

Zigzag Encoder consists of four independent blocks, each of which extracts progressively more complex features from the input. Convolutional filters of smaller sizes from Alex Net are utilized progressively, so that the network learns coarse as well as fine details by the increasing depth of layers. Max Pooling after each convolutional layer reduces the spatial dimensionality, thereby reducing the computational cost while keeping only the most important features. The encoder comprises a Dilated Multi-Scale Inception Network (DiMS-Inception) that enhances multi-scale feature learning without adding additional parameters to increase the receptive field. Skipping pixels would help to capture more context from surrounding areas, which is very important for capturing long-range dependency. This term, "Zigzag", defines the non-linear approach the encoder uses to extract features. The network, therefore, detects different orientations and patterns by zigzagging across various orientations and regions of image or video frames. Each of the four blocks of the ZigZag Encoder specializes in extracting features at different granularities: large spatial patterns using larger convolution kernels in the front blocks and finer, detailed features with smaller kernels in the later blocks.

The Alex Net has 5 convolutional layers and the last 3 layers are fully connected. There are also activation and pooling layers among the layers. Max-pooling layers are coupled in series with convolution layers in the Alex Net architecture. Variable kernel sizes are used by the first convolution layer and subsequently by the max-pooling layers. Since there are four blocks are considered in our work, the last convolution layer of Alex Net is neglected and also the max pooling layers are added with each convolution block unlike Alex Net as in the last three convolution blocks in Alex Net perform only ReLU operation and is not connected with pooling. In our work, Alex Net is used as second encoder which comprises of convolution blocks of 11×11, 5×5 and 3×3. The Alex Net block is connected with a Zigzag path feature learning framework comprises of Inception module with Dilated Convolution layers.

ZigZag Block 1 applies an 11×11 convolutional filter to capture coarse, high-level features that are then followed by Max Pooling to downsample. In this block, more multi-scale features are captured using a Dilated Multi-Scale Inception Network (DiMS-Inception). In ZigZag Block 2, the filter size is reduced to 5×5 for even more refined feature extraction, followed again by Max Pooling and the Dilated Multi-Scale Inception Network (DiMS-Inception) to gather even more contextual information across scales. The size of the filter in ZigZag Block 3 is reduced to 3×3, which is primarily focused on extracting highly detailed local features, although Max Pooling as well as the Dilated Multi-Scale Inception Network (DiMS-Inception) is applied. ZigZag Block 4 is the final block in the network and its backbone is composed of a convolutional filter of size 3×3 because it is very efficient in extracting very fine details and complex patterns similar to the architecture of Alex Net. This block also implements Max Pooling and the Dilated Multi-Scale Inception Network (DiMS-Inception) for multi-scale feature extraction.

3.4 DiMS-inception

The inception module is taken from the Inception A block from the standard Inception V3 architecture in which the conv 3×3 block is divided into three tier dilated convolution blocks with the dilated rate of 4, 8 and 16. The three-tier dilated convolution block is concatenated and is given to the next block. Inception layers are stacked on top of one another with irregular max-pooling layers with a stride of two. They are made up of three convolutional layers: one 1×1 and three 3×3 and five× 5 convolutional layers. The output of each of these layers is a single output vector that serves as the input for the layer that follows. These layers should only be used at higher levels, leaving the other convolutional layers intact, in order to better improve the model memory during training. This architecture's primary advantage is that it allows network width to be increased without uncontrollably raising computational complexity. This design allows the network to capture features at multiple scales without losing spatial information; in later sections, we will see that it proves useful in applications, such as object detection or image and video analysis, when the scale of the objects of interest varies greatly. The DiMS-Inception network architecture is shown in the Figure 4 in which an Inception-based Dilated Convolutional Network with different dilation rates (R), a variation of the popular Inception Network that incorporates dilated convolutions. The architecture comprises of Inception blocks that have been improved using dilated convolutions. Each convolutional path within the block has a different dilation rate (R=4,8,16) this makes it possible for the network to process features at different scale independently. At the same time, 1×1 convolutions are applied for dimensionality decrease of input feature maps, useful to decrease a computational load without losing important information. Dilation is used as a primitive module in this architecture, hence with "holes" or dilation in the convolutional kernels, In order to collect long-range dependencies and contextual information both of which are crucial for segmentation and anomaly detection tasks it expands the receptive field without adding more parameters or sacrificing resolution. The dilated CNN readily enhances the proposed approach, which is based on the 2D convolution layer. Eq. (10) is the mathematical calculations for dilation.

$$y(m,n) = \sum_{i=1}^{m} \sum_{j=i}^{n} x(m + r \times i, n + r \times j) w(i,j) \tag{10}$$

where, $y(m,n)$ denotes the output, $x(m,n)$ denotes the input. The filter where m and n are called length and width, respectively, is illustrated by $w(i,j)$. In the diluted convolution layer Eq. (10), the variable r stands for the dilation rate. When r is used, it refers to the rate of dilation, which is given a distinct number. In case of r is allotted as “1”. A conventional convolution layer will be created from the dilated convolution. Instead of pooling and convolution layer, the Sparse Kernels are used. Additionally, the dilated convolution expands the relevant field without using operators like a convolution layer. This benefit makes the small size kernel $k \times k$ that has been increased in size i.e.,$K + (k - 1)(r - 1)$ where r is a dilation rate that might have several values, such as 2, 3, or 5. Figure 5 depicts the architecture of the DiMS-Inception network.

This architecture begins with a 1×1 convolution layer, which serves two key purposes: preserving the spatial resolution in addition to the dimensionality reduction. It reduces the number of computations and also provides fine spatial details without changing the spatial dimensions of the feature map that reduces computational complexity and allows more detailed computations in the subsequent layers. After this, it is passed through three sets of parallel paths, each with 3×3 convolutions, with different dilation rates (R = 4, 8, 16). The dilation controls the stride rate which is used to space out the kernel elements for the convolution so that it can span a larger receptive field. When R=4, it learns medium-range contextual dependencies, R=8 it learns long-range dependencies, and R=16 it learns a global context without losing spatial resolution. These parallel branches essentially draw out features at varying levels, and their results are combined together either in an interleaved manner through concatenation or in an additive way whereby the final version of the outputs are aggregated representing fine, mid and large scales information of the given input, which is definitely a deeper representation. Downsampling is done by using pool layers, especially average pool layers. This downsizes the spatial dimensions of the feature maps, thereby normalizing the values by averaging over pixels and aids the network to emphasize on significant features, which is helpful in scenarios where global context is important than the local texture. In the final stage 1×1 convolutions are used to make the width and height smaller and make some additional transformations to the features extracted thus making it to suit tasks such as classification or segmentation. The 1×1 convolution helps to fuse multi-scale features together so that the final feature map has an appropriate dimensionality for a given task.

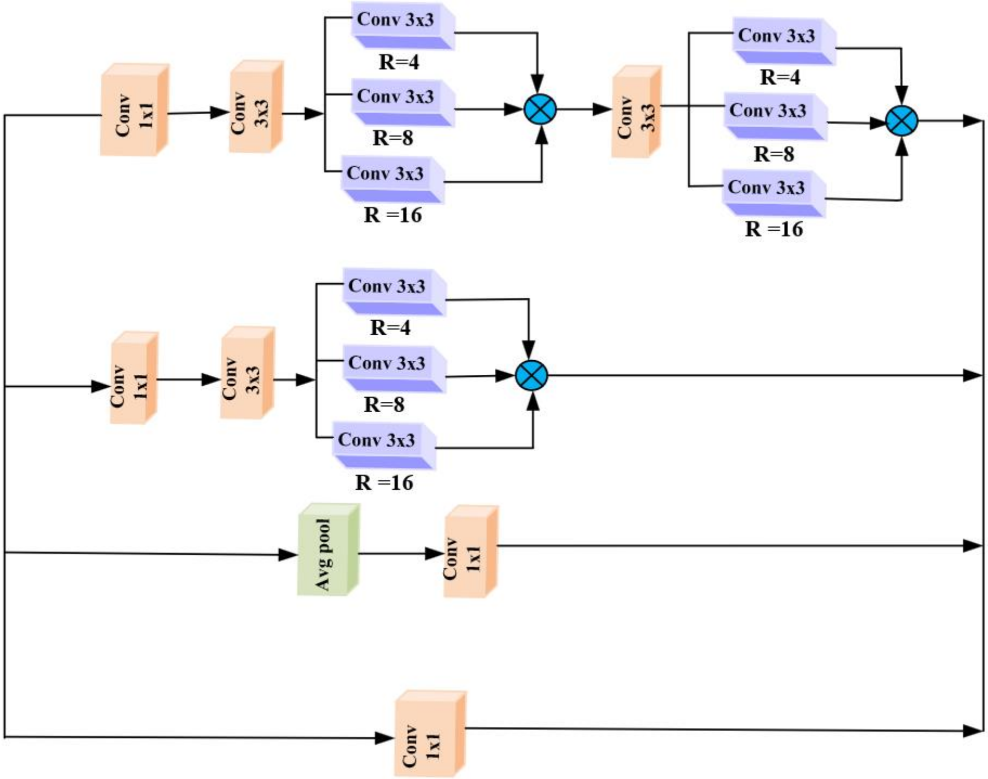


Figure 5. DiMS-Inception network architecture

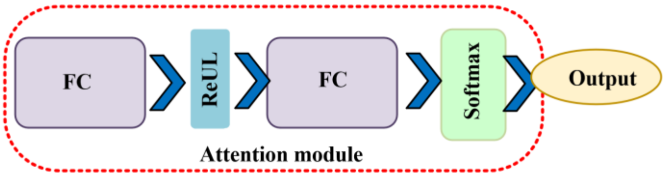


Figure 6. Attention module structure

The architecture of the decoder is U-shaped. The fused feature maps are resized to $\frac{H}{32}, \frac{W}{32}, \frac{D}{32}$ and then processed through an inception-based dilated convolution in the encoder's deepest layer. These feature maps are then subjected to a sequence of upsampling blocks, each of which has a dilated convolution block depending on inception. Through skip connections, these are joined with the encoder's matching feature maps. To further improve the segmentation, the feature maps are combined and then run through an inception-based dilated convolution block. Each upsampling block has an upsampling factor of two and is composed of a ReLU activation function, a normalization layer, and a deconvolutional layer. Finally, segmentation probability maps are calculated using a Sigmoid activation function and an integer value of 1×1×1 3D convolution layer. It uses the attention module to further refine the feature representation. This module improves the model's capacity to detect anomalies by drawing attention to the pertinent areas of the input image. Figure 6 depicts the attention mechanism.

The attention module is intended to improve feature learning since it allows important regions to be given more emphasis. It is comprised of fully connected (FC) layers which helps in mapping the input features into higher-dimensional space and it also has ReLU activation function to introduce non-linearity into the feature space. A second FC layer processes these features and the result is then passed through a SoftMax layer which yields the attention weights. These weights describe the significance of the particular input parts, and the model learns to use essential information. The benefits of attention mechanisms include attention-driven feature learning, model interpretability through understanding the decision-making process, and dynamic attention where the model's focus will change based on the data. The model also enables to extract both high-level and low-level features, enhancing performance by capturing fine-grained details while also understanding broader, more abstract concepts. This attention mechanism enhances detection accuracy by enabling the model to more effectively allocate resources by giving priority to specific areas.

Lastly, the methodology presents explainable AI techniques, such as Local Interpretable Model-agnostic Explanations (LIME) for fine-tuning individual predictions and Grad-CAM (Gradient-weighted Class Activation Mapping) for creating heatmaps that explain the regions influencing the model's predictions. Figure 7 illustrates the visualization of GRAD-Cam and LIME. Figure 8 shows the real-time visualization

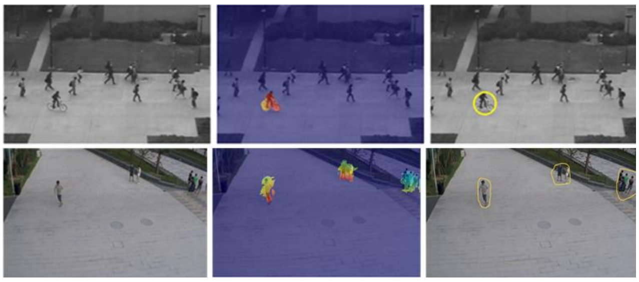


Figure 7. GRAD-Cam and LIME visualization

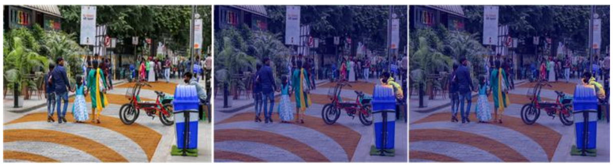


Figure 8. Real time GRAD-Cam and LIME visualization

4. RESULTS AND DISCUSSION

This section discusses the experiment's results using evaluation metrics such as precision, precision of the F-measure, sensitiveness, specificity, NPV, FPR, and FNR. The suggested X2_PDDVnet for detecting anomalies in events. The suggested model is implemented in Python. The evaluation results are contrasted with those of several other methods, including the proposed X2_PDDVnet, ViT, CNN-Bi-LSTM, Swin Transformer, and U-Net.

4.1 Dataset description

Three datasets, including Avenue, UCSD Anomaly Detection, and Shanghai Tech, are used to assess the effectiveness of the suggested approach. These datasets were taken at a specific location using a static camera. The testing video of the aforementioned dataset retains both normal and aberrant occurrences, whereas the training films only contain normal events.

4.1.1 UCSD Anomaly Detection

Dataset 1

(http://www.svcl.ucsd.edu/projects/anomaly/dataset.htm).

The UCSD Anomaly Detection Dataset was gathered using a stationary camera that was situated at a height that provided a view of pedestrian routes. The pedestrian density on the pathways ranged from dense to sparse. The only individuals in the film are passersby in a normal setting. Unusual pedestrian movement patterns or the movement of non-pedestrian objects along the paths can result in strange events. Anomalies are common and include people crossing a walkway or in the grass around it, bikers, skaters, and small carts. There were also a few incidences involving wheelchair users. Since none of the anomalies were manufactured for the dataset's assembly, they are all spontaneously occurring. Two subsets of the data were created, each of which represented a distinct scene. The 200-frame video footage that was captured from each scene was divided into different clips. Peds1: video featuring throngs of people moving either toward or away from the camera, along with a small amount of perspective distortion. There are 34 teaching and 36 testing video samples. Peds2: situations where the camera plane and pedestrian movement are parallel. contains sixteen teaching video examples and twelve testing video examples. Each clip's ground truth annotation includes a binary flag for each frame that indicates whether an abnormality is present or not. Furthermore, manually created pixel-level binary masks are included for a subset of 10 films for Peds1 and 12 clips for Peds2, identifying the areas with anomalies. This will allow performance to be assessed in terms of algorithms' capacity to locate anomalies.

4.1.2 Avenue Dataset

Dataset 2

(https://www.cse.cuhk.edu.hk/leojia/projects/detectabnormal/dataset.html).

The Avenue Dataset includes 21 testing and 16 training video segments. Normal circumstances are captured in the training videos. Videos of tests show both typical and unusual occurrences. The videos, which contain 30652 frames (15328 training, 15324 testing), were shot on CUHK Campus Avenue. People are frequently seen wandering about the subway entrance and stairwell. Unusual events, however, include individuals hopping, tossing bags, lingering, hurrying and going in the wrong direction, and so forth. Among the challenges of the dataset are camera movement and some abnormalities in the training samples.

4.1.3 Shanghai Tech Dataset

Dataset 3

(https://www.kaggle.com/datasets/alterralaniakea/shanghaitech-anomalous-behaviour).

This dataset is the largest for the unsupervised VAD and consists of 437 video clips with an 856 × 480 pixel frame size. 318 training and 107 testing films from 13 different backdrops are included in this dataset. The movies are gathered by changing the camera angles and lighting to broaden the dataset's diversity and make it more representative of real-world situations. This dataset contains 130 real-world anomalous occurrences, such as vehicles, motorcycles, bicycles, skateboards, fighting, chasing, and jumping.

4.2 Performance metrics

The performance metrics employed to evaluate the proposed model are given below.

Accuracy: The ratio of samples that were successfully identified to all samples is used to determine accuracy.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

Precision: Out of all the samples that were predicted to be positive, precision quantifies the proportion of samples that they were correctly identified as positive.

$$Precision = \frac{TP}{TP + FP} \tag{12}$$

Sensitivity: The percentage of true positive samples that were correctly classified as positive is measured by sensitivity.

$$Sensitivity = \frac{TP}{TP + TN} \tag{13}$$

Specificity: Specificity is determined by how many actual negative samples are correctly categorised as negative.

$$Specificity = \frac{TN}{TN + FP} \tag{14}$$

F-Measure: It is the harmonic mean of recall and precision.

$$F - Measure = \frac{2 * precision * recall}{Precision + recall} \tag{15}$$

FPR: The FPR calculates the proportion of really negative samples that were incorrectly regarded as positive.

$$FPR = \frac{FP}{FP + TN} \tag{16}$$

FNR: The percentage of true positive samples that were incorrectly classified as negative is determined by FNR.

$$FNR = \frac{FN}{TP + FN} \tag{17}$$

MCC: MCC spans from -1 to +1 and integrates data regarding true and false positives and negatives into a single value, where +1 denotes a perfect classification, 0 denotes random categorization, and -1 denotes the full discrepancy between prediction and observation.

$$MCC = \frac{((TP * TN) - (FP * FN))}{\sqrt{((TP + FP) * (TP + FN) * (TN + FP) * (TN + FN))}} \tag{18}$$

NPV: Out of all the samples that were predicted to be negative, NPV determines the proportion of actual negative samples that are accurately identified as such.

$$NPV = \frac{TN}{TN + FN} \tag{19}$$

where,

FP (False Positives) - Number of incorrectly classified negatively classified samples,

FN (False Negatives) - Number of incorrectly classified positively classified samples,

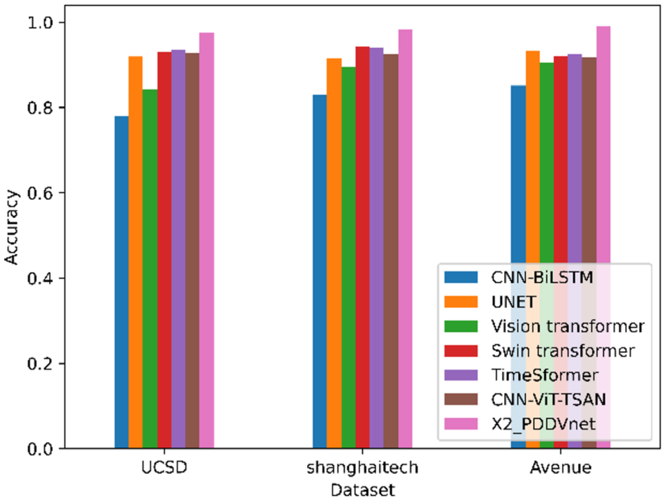
TP (True Positives) - Number of correctly classified positively classified samples,

TN (True Negatives) - Number of correctly classified negatively classified samples.

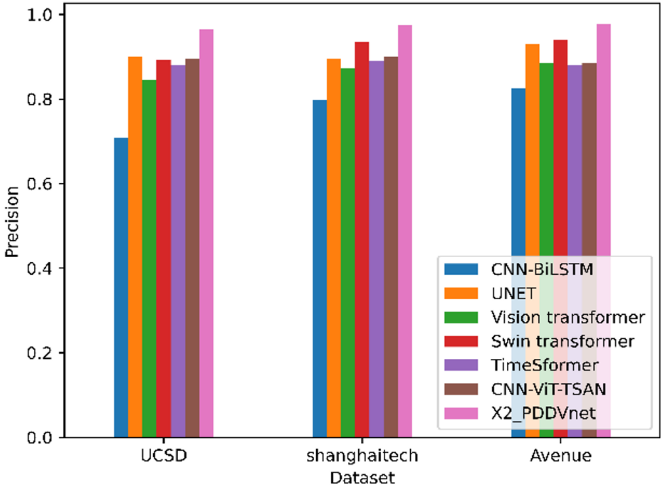
4.3 Performance analysis of the proposed model

The performance analysis is done on various models on three datasets utilized for anomaly detection: Dataset 1 (UCSD Anomaly Detection Dataset), Dataset 2 (Avenue Dataset for Abnormal Event Detection), and Dataset 3 (ShanghaiTech Dataset). The graphical analysis for the various performance metrics like accuracy, precision, sensitivity and specificity is shown in the Figures 9(a)-(d).

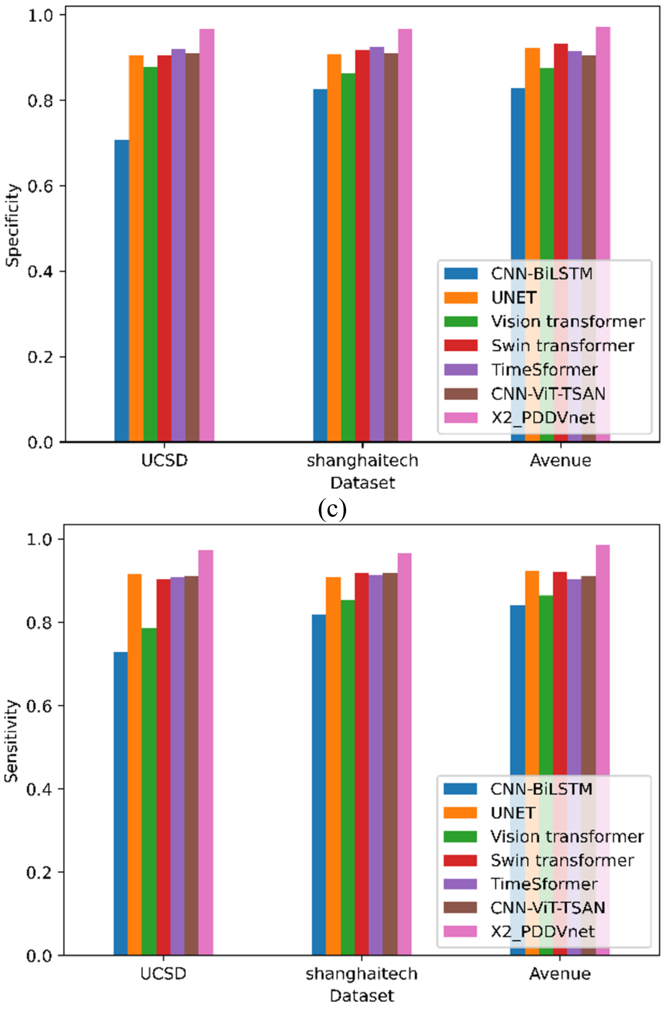
Each model's performance demonstrates its effectiveness in identifying anomalies, with higher accuracy values indicating improved anomaly detection capability. CNN-BiLSTM achieved moderate accuracy on all the datasets. Accuracy score of CNN-BiLSTM is 0.7794 on Dataset 1, 0.83 on Dataset 2, and 0.8512 on Dataset 3. UNET gained more accuracy values as it reached 0.9191 on Dataset 1, 0.915 on Dataset 2, and 0.9323 on Dataset 3. The accuracy value of ViT model is more than CNN-BiLSTM in all datasets. Accuracy value for Datasets 1, 2, and 3 is 0.8422, 0.895, and 0.9054, respectively. However, the accuracy is less than UNET. The proposed X2_PDDVnet achieved the highest accuracy of any of the models for any of the datasets and scored 0.9763 on dataset 1, 0.983 on dataset 2, and 0.9898 on dataset 3. The dual-path architecture of this proposed model, in which it would integrate the ViT by a ZigZag-enhanced AlexNet, guarantees maximum feature extraction across any scale. CNN-BiLSTM is moderately accurate with lesser scores of 0.7087, 0.798, and 0.825 in Dataset 1, 2, and 3 implying its failure to acquire tough time dependencies, thus increasing its false positives.UNET results higher precision than CNN-BiLSTM. The ViT performs better than CNN-BiLSTM because it uses the global attention.



(a)



(b)



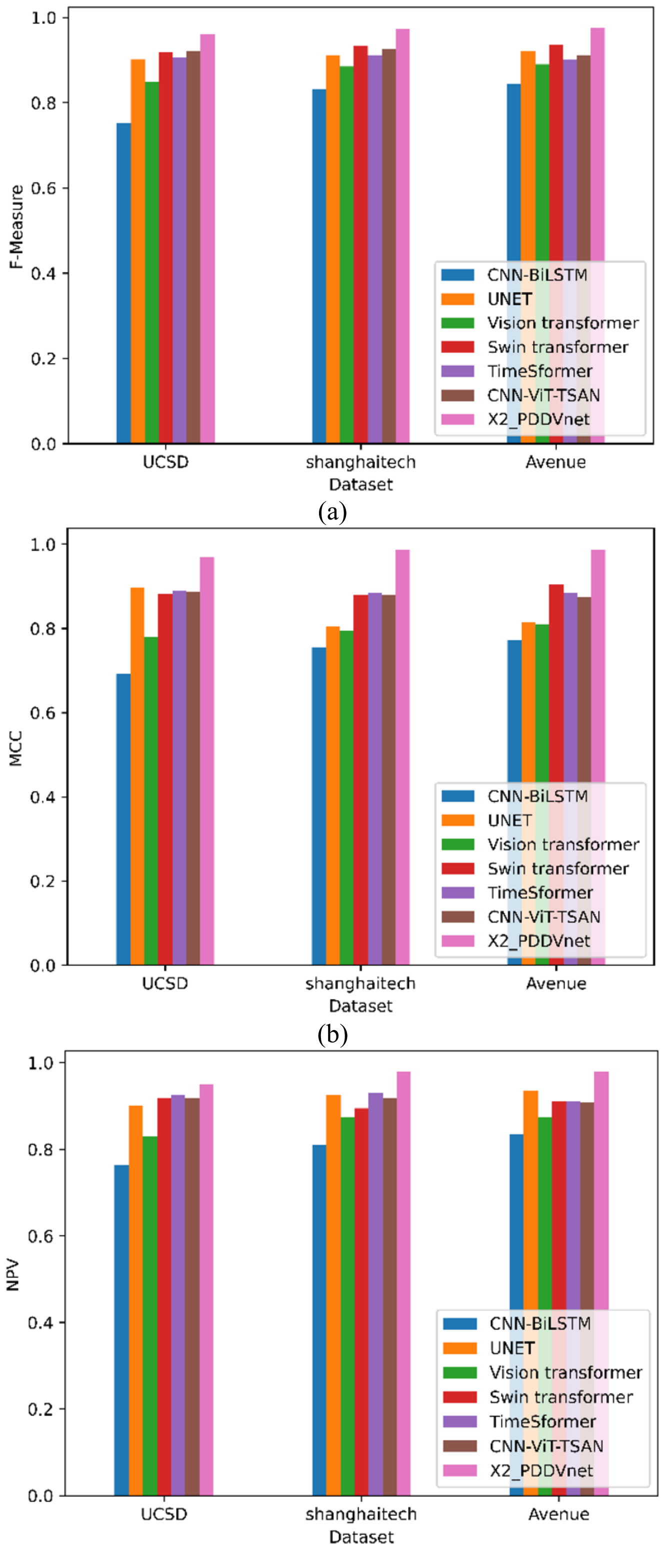
(c)

(d)

Figure 9. Performance evaluation of (a) Accuracy, (b) Precision, (c) Sensitivity, (d) Specificity

Swin Transformer, using multi-scale spatial processing, is efficient in precision for all data sets, especially for the Avenue and ShanghaiTech ones. The proposed X2_PDDVnet model achieves the highest precision scores on all datasets with values of 0.9643, 0.976, and 0.977.

CNN-BiLSTM has a relatively moderate sensitivity but ranks the lowest, probably because it is not able to capture much sequential pattern. UNET shows improved sensitivity scores of 0.9174, 0.91, and 0.925 and is better than CNN-BiLSTM. The ViT improves further with values of 0.7854, 0.853, and 0.865 for Dataset 1, 2 and 3, which utilizes global attention but lacks localized feature focus. The Swin Transformer achieves strong sensitivity, as it has a hierarchical, multi-scale structure. The highest sensitivity scores are achieved by X2_PDDVnet with scores of 0.9743, 0.967 and 0.9856 for Datasets 1, 2 and 3 respectively. CNN-BiLSTM has specificity of 0.7081, 0.826, and 0.83 for Dataset 1,2 and 3. UNET has better specificity and has scores of 0.9065, 0.907, and 0.9224. ViT outperforms CNN-BiLSTM because the former makes use of its global attention mechanism, scoring specificities of 0.8785, 0.864, and 0.875; but it is not that specific as UNET. The Swin Transformer high scores are 0.905, 0.918, and 0.932 for Dataset 1, 2 and 3, capitalizing on its hierarchical structure to significantly outperform other models for distinguishing normal events. Specificity for Dataset 1,2 and 3 is at a maximum score of 0.9665, 0.968, and 0.9725 with X2_PDDVnet. The graphical analysis for the various performance metrics like F-measure, MCC and NPV is shown in the Figures 10(a)-(c).



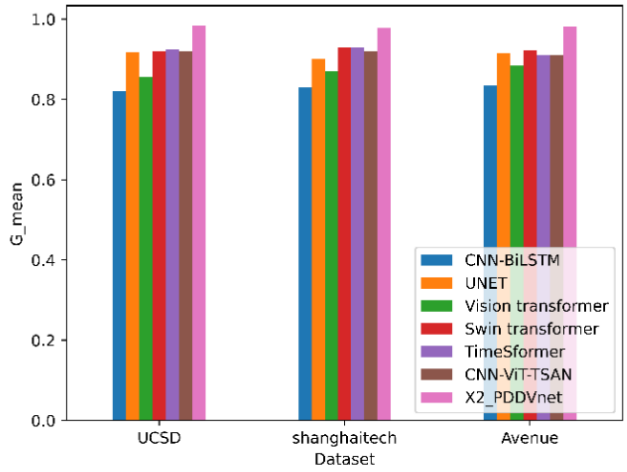
(a)

(b)

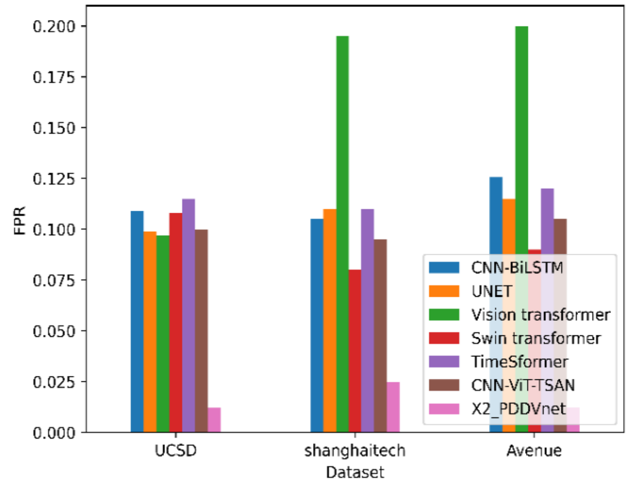
(c)

Figure 10. Performance evaluation of (a) F-measure, (b) MCC and (c) NPV

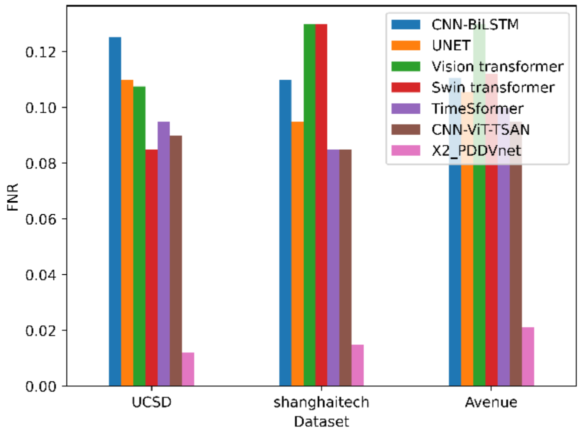
CNN-BiLSTM achieves relatively balanced F-measure values 0.7525, 0.832 and 0.843 respectively, with the lowest scores among all models because they are unable to capture a lot of complex dependencies in the network. UNET performs better than CNN-BiLSTM with scores of 0.9021, 0.91, and 0.9215. Then, ViT improved scores from CNN-BiLSTM to 0.8498, 0.885, and 0.89 by supporting its global attention mechanism. Swin Transformer scores 0.918, 0.934, and 0.936, due to its hierarchical, multi-scale structure. X2_PDDVnet gets the highest F-measure values, 0.9612, 0.973, and 0.9756 for Dataset 1, 2 and 3 respectively. CNN-BiLSTM achieves MCC values at 0.6921, 0.755, and 0.7723 but ranks low due to weak capture of the intricate relationships. On the other hand, UNET surpasses this through an MCC value of 0.8976, 0.805, and 0.8156. ViT 's results are better as they surpassed the values set by CNN-BiLSTM at 0.7809, 0.794, and 0.81. The Swin Transformer obtains the highest MCC values, 0.883, 0.88, and 0.905, exploiting its hierarchical multi-scale structure to effectively capture spatial features. X2_PDDVnet obtains the highest MCC values, with 0.9689, 0.988, and 0.9878 for Dataset 1, 2 and 3 respectively. For the UCSD dataset, CNN-BiLSTM attained 0.7632 of NPV, UNET was even more promising with 0.9009, while the ViT was even better at 0.83. The Swin Transformer offers even more encouraging results at an accuracy rate of 0.919. However, X2_PDDVnet gives the NPV at 0.9511, thus showing more reliability. In Avenue Dataset, CNN-BiLSTM is at 0.81, UNET reaches 0.925, ViT achieves 0.875, Swin Transformer scores 0.895, but X2_PDDVnet again surpasses all with 0.979, showing great differentiation of normal events. In ShanghaiTech, CNN-BiLSTM's NPV is 0.8352, UNET scores 0.9354, ViT maintains 0.875, Swin Transformer achieves 0.912, and again X2_PDDVnet leads with 0.9789 of NPV. The graphical analysis for the various performance metrics like G-mean, FPR and FNR is shown in the Figures 11(a)-(c).



(a)



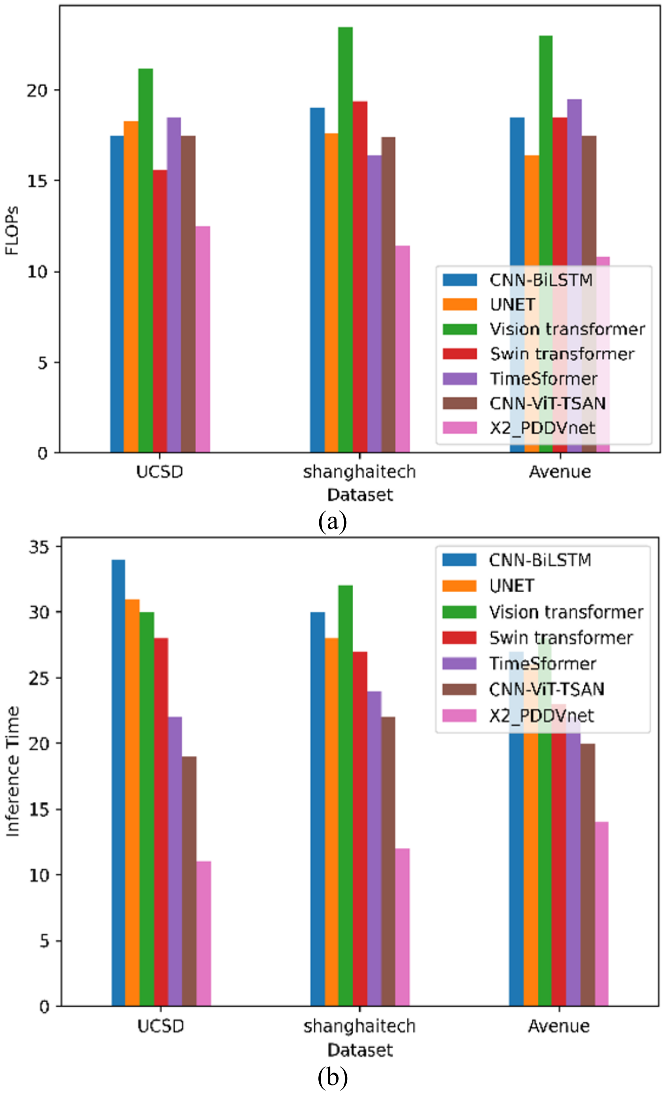
(b)



(c)

Figure 11. Performance evaluation of (a) G-mean, (b) FPR and (c) FNR

In the UCSD Anomaly Detection Dataset, X2_PDDVnet outperforms with a Geometric mean (G-mean) of 0.9843, followed by the Swin Transformer at 0.919 and UNET at 0.9175, which is strong reliability. The ViT scores 0.8565, and CNN-BiLSTM has the lowest score at 0.8212. In the Avenue Dataset, X2_PDDVnet again leads with a G_mean of 0.978, followed by the Swin Transformer at 0.93 and UNET at 0.9. The CNN-BiLSTM is lower with a score of 0.83 for the ShanghaiTech Dataset. X2_PDDVnet attained highest G_mean with a 0.982 value in the ShanghaiTech Dataset while Swin Transformer and UNET are at 0.9234 and 0.9154 respectively, followed by ViT 's score of 0.885, and the CNN-BiLSTM acquired lowest score of 0.8348. CNN-BiLSTM and Swin Transformer report moderate FPRs as 0.109 and 0.108 for UCSD, while UNET and ViT show a slight improvement at 0.0987 and 0.0968, respectively. X2_PDDVnet performs the best with the lowest FPR of 0.0121, which shows higher accuracy on UCSD dataset. On the Avenue Dataset, CNN-BiLSTM scores 0.105, UNET scores 0.11, and ViT shows an extremely high FPR at 0.195. With 0.08, Swin Transformer does not perform well; however, the lowest FPR is reported by X2_PDDVnet with 0.025. On the ShanghaiTech dataset, ViT reports the highest FPR of 0.2, while CNN-BiLSTM, UNET, and Swin Transformer report moderate FPRs of 0.1256, 0.115, and 0.09, respectively. X2_PDDVnet outperforms all others with a low FPR of 0.0123. In the UCSD Anomaly Detection Dataset, X2_PDDVnet has the best FNR of 0.0121, which makes it highly accurate in its anomaly detection capabilities, with the Swin Transformer close behind at 0.085. The UNET and ViT are then better than the CNN-BiLSTM, by having FNRs, 0.1098 and 0.1076, respectively. Regarding the Avenue Dataset, X2_PDDVnet attained a low FNR of 0.015, while the CNN-BiLSTM is moderate with an FNR of 0.11. Transformer and Swin Transformer show high FNR of 0.13. In other words, more than 0.13 cases are missed in its detection. In ShanghaiTech Dataset, X2_PDDVnet is highly rated with an FNR of 0.0212; CNN-BiLSTM and Swin Transformer have almost the same outcomes at about 0.11. Overall, X2_PDDVnet has a better performance on all the datasets, making it a robust model for anomaly detection, while CNN-BiLSTM needs to improve its capabilities. Figure 12 shows the flops and interference time.



(a)

(b)

Figure 12. (a) Flops and (b) Interference time

4.4 Ablation study analysis

The ablation study is investigated on the proposed model for the three datasets for Encoder 1, Encoder 2 and Combined proposed model. The ablation study on the UCSD Anomaly Detection Dataset show significant improvements when combining the ViT Encoder and ZigZag Encoder with a decoder, highlighting the effectiveness of this integrated approach is illustrated in the Table 1.

The combined proposed model scored 0.9763, much higher than the individual encoders, which scored 0.93 and 0.85, respectively in terms of accuracy. Similarly, precision was at its highest in the combined model at 0.9643, indicating more effectiveness in identifying true positives compared to the encoders alone. Sensitivity improved dramatically as the combined model reached 0.9743, hence better detecting true anomalies without increasing false negatives. Specificity also showed improvement as the combined model achieved 0.9665, with better accuracy in identifying true negatives. F-measure revealed a huge improvement in the combined method at 0.9612. The MCC also emphasized on the robustness of the merged model, and it resulted in a score of 0.9689; the ViT Encoder scores were 0.905 and 0.7 for the ZigZag Encoder. The NPV was 0.9511 for the merged model meaning that the classification for the non-anomalous data results in a higher precision; the FPR in case of the merged model considerably reduced to 0.0121 with the FNR at 0.0121 as compared to actual anomalies not being missed. Lastly, the Gmean was at its peak for the combined model with a value of 0.9843, which means better all-around performance since it equilibrates sensitivity and specificity. The graphical analysis for the comparison of individual and combined approach is depicted in the Figure 13.

The efficacy of this combined approach is highlighted by the ablation study on the Avenue Dataset in the Table 2, which shows significant gains when integrating the ViT Encoder and ZigZag Encoder with a decoder. Table 2 presents a comparison analysis of the Avenue Dataset.

Table 1. Comparison analysis on the UCSD anomaly detection dataset

Metric	Encoder 1 (ViT Encoder)	Encoder 2 (ZigZag Encoder)	Encoder 1 + Encoder 2 + Decoder
Accuracy	0.93	0.85	0.9763
Precision	0.912	0.84	0.9643
Sensitivity	0.9098	0.78	0.9743
Specificity	0.912	0.87	0.9665
F-Measure	0.927	0.81	0.9612
MCC	0.905	0.7	0.9689
NPV	0.925	0.79	0.9511
FPR	0.109	0.13	0.0121
FNR	0.0985	0.22	0.0121
G-mean	0.9126	0.876	0.9843

Table 2. Comparison analysis on the Avenue Dataset

Metric	Encoder 1 (ViT Encoder)	Encoder 2 (ZigZag Encoder)	Encoder 1 + Encoder 2 + Decoder
Accuracy	0.925	0.86	0.983
Precision	0.918	0.9	0.976
Sensitivity	0.917	0.8	0.967
Specificity	0.9182	0.92	0.968
F-Measure	0.9176	0.84	0.973
MCC	0.9264	0.71	0.988
NPV	0.9258	0.83	0.979
FPR	0.098	0.08	0.025
FNR	0.093	0.21	0.015
Gmean	0.945	0.8455	0.978

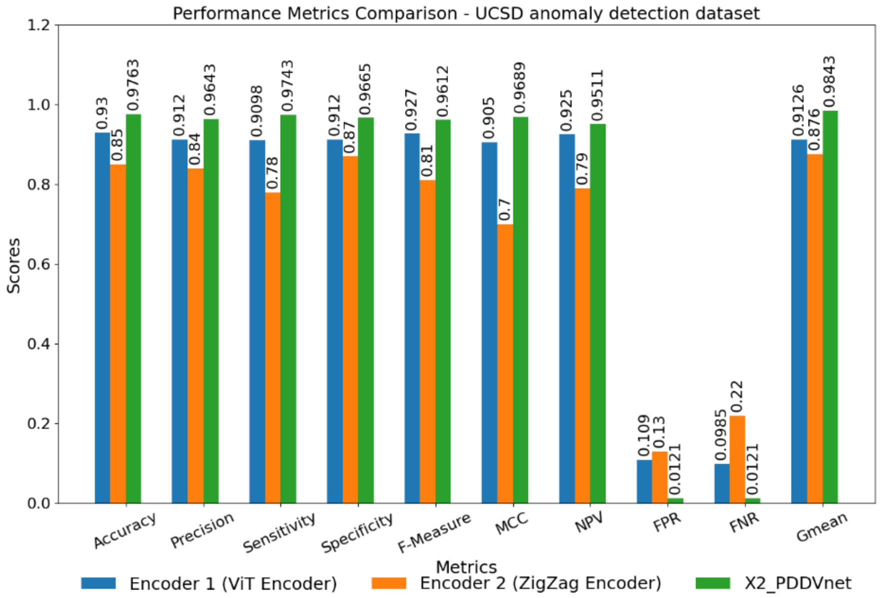


Figure 13. Graphical analysis on the UCSD anomaly detection dataset

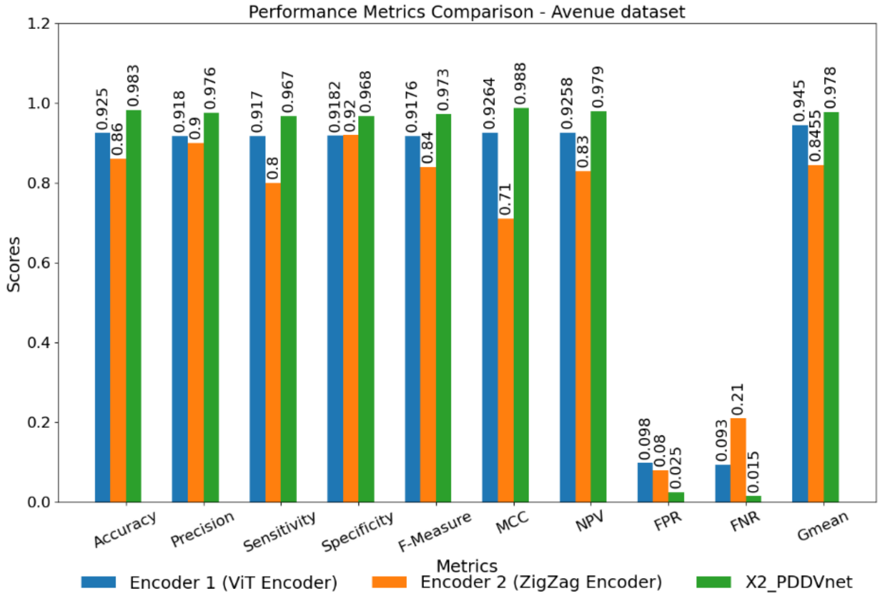


Figure 14. Graphical analysis on the Avenue dataset

Table 3. Comparison analysis on the ShanghaiTech dataset

Metric	Encoder 1 (ViT Encoder)	Encoder 2 (ZigZag Encoder)	Encoder 1 + Encoder 2 + Decoder
Accuracy	0.95	0.84	0.9898
Precision	0.9	0.86	0.9778
Sensitivity	0.915	0.79	0.9856
Specificity	0.925	0.84	0.9725
F-Measure	0.896	0.82	0.9756
MCC	0.915	0.68	0.9878
NPV	0.92	0.81	0.9789
FPR	0.205	0.16	0.0123
FNR	0.135	0.21	0.0212
G-mean	0.902	0.8443	0.982

The accuracy of the combined model was 0.983, while the individual encoders achieved accuracy scores of 0.925 for ViT Encoder and 0.86 for ZigZag Encoder for Avenue Dataset. It could be because the model has augmented its ability to classify the anomalies with higher accuracy. The precision also increased from 0.918 for the ViT Encoder and 0.9 for the ZigZag Encoder to 0.976 in the combined model, which implies better actual positive cases and fewer false positives. Sensitivity increased from 0.917 of ViT Encoder and 0.8 with ZigZag Encoder to the combined approach of 0.967, reflecting how the model is better able to identify true anomalies with less missed detection. With a specificity of 0.968, the combined model outperformed the individual encoders, which had scores = 0.9182 and 0.92, respectively. Furthermore, the F-Measure rose to 0.973, demonstrating a solid balance between both accuracy and recall, up from the values: 0.9176 for ViT Encoder and 0.84 for ZigZag Encoder. MCC also showed good improvement, rising up to 0.988 from the values of 0.9264 of ViT Encoder and 0.71 at ZigZag Encoder, indicating the stability of the combined approach. NPV increased to 0.979 as compared to 0.9258 of ViT Encoder and 0.83 of ZigZag Encoder which increased the accuracy of prediction for non-anomalous instances. FPR reduced to 0.025 from 0.098 of ViT Encoder and 0.08 at ZigZag Encoder, and FNR decreased to 0.015, but for individual encoders, FNR was 0.093 and 0.21 respectively, hence showing more accuracy in detection of actual anomalies. Lastly, the Gmean increased from 0.945 for the ViT Encoder to 0.8455 for the ZigZag Encoder and increased to 0.978 in the combined model, further proving the better performance of the integrated approach in the Avenue Dataset. The graphical analysis for the comparison of individual and combined approach in Avenue Dataset is depicted in the Figure 14.

The efficacy of this combined approach is highlighted by the ablation study on the ShanghaiTech dataset is illustrated in Table 3, which shows significant gains when integrating the ViT Encoder and ZigZag Encoder with a decoder. Table 3 presents a comparison analysis of the ShanghaiTech dataset.

In the Shanghai Tech Dataset, the combined model scored an impressive accuracy of 0.9898 compared to individual encoders with 0.95 for the ViT Encoder and 0.84 for the ZigZag Encoder. Precision was increased to 0.9778 in the combined model from 0.9 and 0.86 for the ViT and ZigZag Encoders, which illustrates an increased efficiency for the identification of true positives and lowered false positives. The sensitivity improved to 0.9856, which means that the model was much more sensitive to true anomalies and had fewer missed detections, which is higher than ViT Encoder's 0.915 and ZigZag Encoder's 0.79. Specificity was very high at 0.9725 for the combined model compared to the individual encoders' scores of 0.925 and 0.84, which reflected an improved performance in correctly identifying true negatives. In addition, F-Measure increased up to 0.9756 and MCC significantly up to 0.9878, which indicates the strength of the combined approach. The NPV was increased to 0.9789, indicating the stronger ability to predict the cases that are not anomalous. Moreover, the value of FPR is reduced up to 0.0123 and FNR to 0.0212, which shows more abilities for the correct detection of negative and positive cases. More precisely, the Gmean of 0.982 further underlines better performance of the integrated approach with regard to better sensitivity and specificity balance. The graphical analysis for the comparison of individual and combined approach in Shanghai Tech Dataset is depicted in the Figure 15.

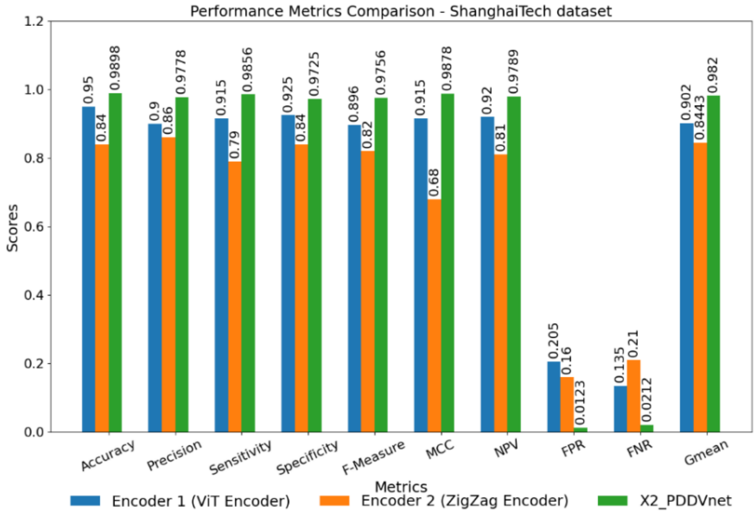
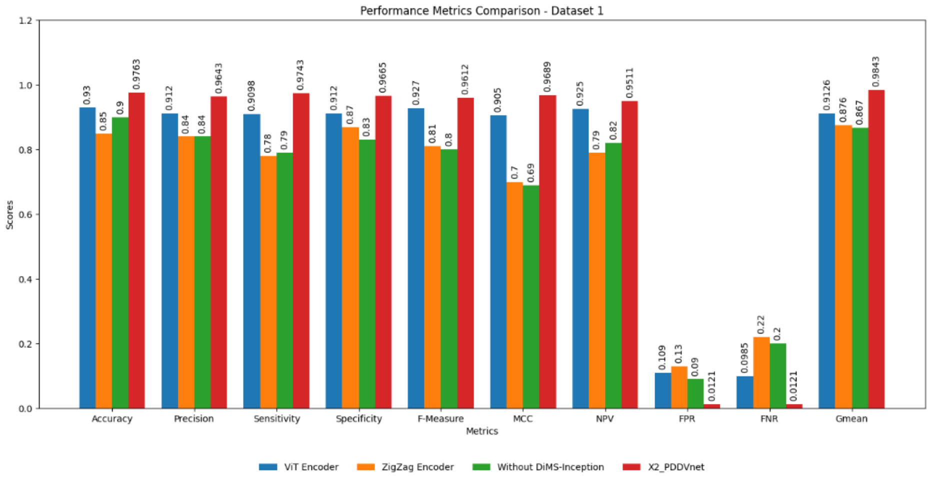
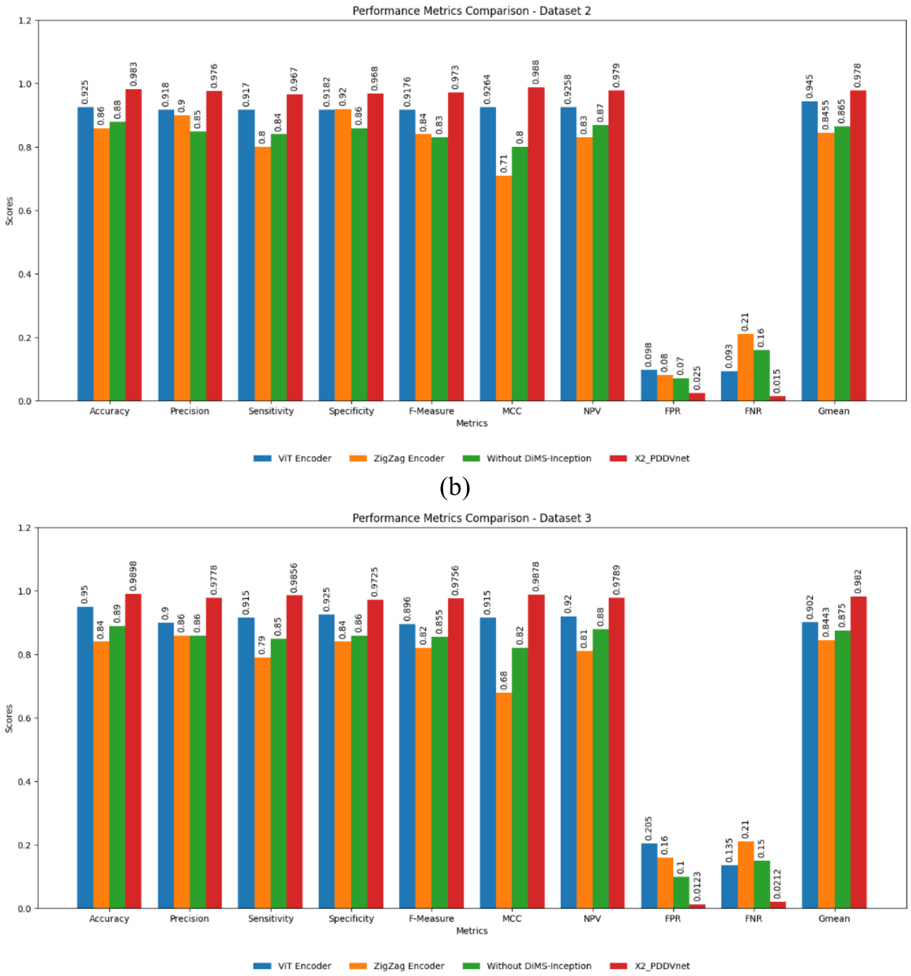


Figure 15. Graphical analysis on the Shanghai Tech Dataset



(a)



(b)

(c)

Figure 16. Performance metrics: (a) Dataset 1, (b) Dataset 2, (c) Dataset 3

The performance of four encoder structures (VIT Encoder, Zigzag Encoder, Without Shift-Inception, and IQ-PDVNet) for different evaluation metrics. IQ-PDVNet generally has high performance, achieving maximum or second-highest values in metrics such as Accuracy, Precision, Sensitivity, and Gmean. VIT Encoder also shows competitive performance in multiple metrics for both datasets. Zigzag Encoder and Without Shift-Inception have more inconsistent performance, with some trailing the other two encoders based on metric and dataset. IQI and HRA metrics have low scores for all encoders on both datasets, suggesting they may be more challenging parts of the task. The Gmean metric indicates good overall performance of IQ-PDVNet. Although there are individual variations between datasets, the overall trend in relative performance among IQ-PDVNet and VIT Encoder is largely comparable, suggesting that architectural decisions in IQ-PDVNet contribute to strong performance across different datasets and tasks.

Figure 16 illustrates the performance metrics Accuracy, Precision, Sensitivity, Specificity, F-Measure, MCC, NPV, FNR, FPR, and G-Mean evaluated on Dataset 1, Dataset 2, and Dataset 3 using four models: ViT Encoder, ZigZag Encoder, Without DiMS Inception, and X2_PDDVNet

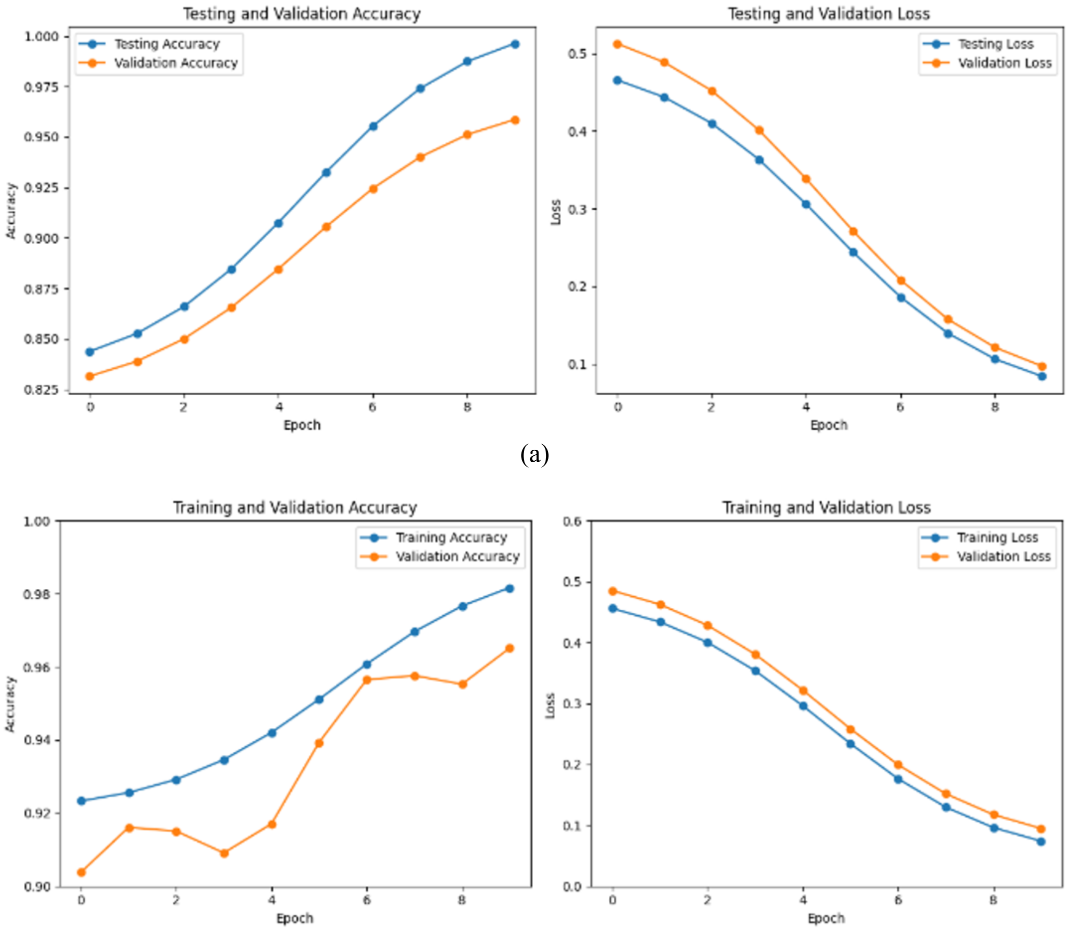
4.5 Comparing the performance of different loss function models

The performance metrics of several models applied to a segmentation task are shown in Table 3, which is assessed using three distinct loss functions: Dice loss, focal loss, or a mix of the two (Focal + Dice Loss). Accuracy, Mean Average Precision (mAP), and Mean Intersection over Union (mIoU) are among the parameters that are presented. Table 4 presents a performance comparison of various loss functions.

For the U-Net model, the mIoU is 0.89, mAP is 0.86, and accuracy is 90.80% using Focal loss. If Dice Loss is used, the performance is a bit worse, with an mIoU of 0.88, mAP of 0.85, and an accuracy of 90.03%. However, the performance of U-Net gets enhanced to a much greater extent when both loss functions are used, with an mIoU of 0.90, mAP of 0.90, and an accuracy of 92.76%. A strong competing model is also developed for the ViT by presenting an mIoU of 0.90, mAP of 0.91, and an accuracy of 92.40% using Focal Loss. The model develops its performance a little while applying Dice Loss as indicated in the mIoU at 0.91, mAP at 0.90, and an accuracy of 92.77%. Again, the best performer for this model is Focal Loss with Dice Loss, as mIoU was 0.90, mAP was 0.92, and an accuracy of 93.78%. Swin Transformer with Focal Loss brought out the result of an mIoU of 0.923, mAP of 0.906, and an accuracy of 93.55%. Again, better performance is offered by the use of Dice Loss: the mIoU reached 0.9353, mAP as 0.9213, and an accuracy of 92.80%. The further combination of the both the loss functions results in the performance to achieve an mIoU of 0.943, mAP of 0.9323, and the accuracy as 93.44%. However, CNN-BiLSTM had a lower overall score with Focal Loss, an mIoU of 0.8376, a mAP of 0.8972, and accuracy of 89.47% were obtained. Using Dice Loss, a small improvement is seen with mIoU of 0.8456, mAP of 0.8866, and accuracy of 88.38%. Focal + Dice Loss had a marginally improved mIoU of 0.8645, mAP of 0.856, and accuracy of 89.53%. The X2_PDDVnet architecture has proved to be the best since it provides results of an mIoU of 0.9435, mAP of 0.9543, and accuracy of 97.44% Focal Loss. It was even observed that the usage of Dice Loss resulted in mIoU of 0.9536, mAP of 0.9654, and accuracy of 97.34%. Best performance is found in combination of both with the impressive results, including a maximum IoU of 0.9642, and mAP of 0.91, with 98.42% of accuracy. Figure 17 (a), (b) and (c) depict the training and validation accuracy and loss graphs for the three datasets.

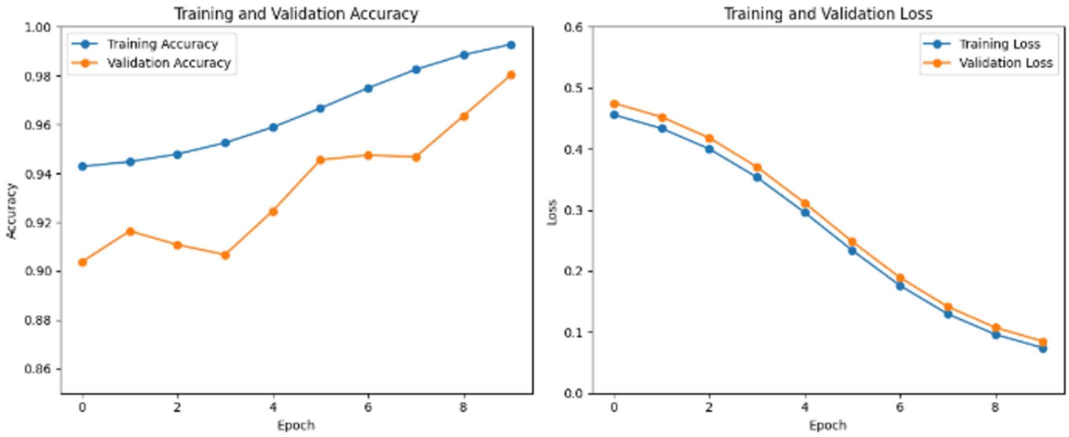
Table 4. Performance comparison of various loss function

Model	Loss Function	mIoU	mAP	Accuracy
U-Net	Cross-Entropy	0.85	0.8	89.50%
U-Net	Dice Loss	0.87	0.84	91.03%
U-Net	Cross-Entropy + Dice Loss	0.89	0.86	90.80%
Vision Transformer	Cross-Entropy	0.88	0.89	92.10%
Vision Transformer	Dice Loss	0.92	0.91	91.89%
Vision Transformer	Cross-Entropy + Dice Loss	0.9	0.91	92.40%
Swin Transformer	Cross-Entropy	0.91	0.89	93.00%
Swin Transformer	Dice Loss	0.93	0.92	92.68%
Swin Transformer	Cross-Entropy + Dice Loss	0.94	0.93	93.44%
CNN-BiLSTM	Cross-Entropy	0.83	0.88	88.56%
CNN-BiLSTM	Dice Loss	0.85	0.87	88.38%
CNN-BiLSTM	Cross-Entropy + Dice Loss	0.86	0.86	89.47%
TimeSformer	Cross-Entropy	0.91	0.92	93.30%
TimeSformer	Dice Loss	0.93	0.93	93.80%
TimeSformer	Cross-Entropy + Dice Loss	0.94	0.93	93.50%
CNN-ViT-TSAN	Cross-Entropy	0.87	0.89	92.45%
CNN-ViT-TSAN	Dice Loss	0.91	0.92	91.70%
CNN-ViT-TSAN	Cross-Entropy + Dice Loss	0.92	0.93	92.80%
Proposed	Cross-Entropy	0.94	0.95	97.12%
Proposed	Dice Loss	0.95	0.97	97.34%
Proposed	Cross-Entropy + Dice Loss	0.96	0.97	98.42%



(a)

(b)



(c)

Figure 17. Training validation accuracy and loss for UCSD dataset, Avenue Dataset, and Shanghai dataset

Across the UCSD, Avenue, and Shanghai datasets, the model demonstrates consistent improvement in both accuracy and loss metrics, which reflects the robust learning and generalization capability of the model. For the UCSD dataset, the testing accuracy starts at 0.850 and increases to 0.990. Validation accuracy increases from 0.825 to approximately 0.950, with strong generalization to unseen data. Testing and validation loss of UCSD also improve progressively. Their values decrease from 0.450 to 0.100 and from 0.500 to 0.150, respectively. The training accuracy begins at 0.920 and peaks at 0.980, while validation accuracy follows, beginning around 0.900 and peaking at 0.960, indicating good learning and generalization. Training and validation loss both steadily decrease, from 0.450 to 0.100 and 0.500 to 0.150, respectively, indicating decreased error in the training and validation data. Similarly, for the Shanghai dataset, the accuracy of the training set is from 0.920 to 0.980, and validation accuracy is from 0.900 to 0.960 and demonstrates strong generalization ability. The training loss is reduced from 0.450 to 0.100 and the validation loss from 0.500 to 0.150, showing the errors are further minimized. At the end of training, the model can attain near-peak accuracy values at 0.990 for testing and 0.960 for validation; it also reduces losses consistently to about 0.100 on testing and 0.150 on validation, which hints towards the high performance and generalization capability of the model.

5. CONCLUSION

A new video anomaly detection approach is proposed in this paper, known as X2_PDDVnet, which is an integration of a dual-path architecture based on a ViT and an AlexNet-based CNN with the enhancement of a ZigZag path learning method. This dual-path setup enables the model to capture effective global and local spatial patterns uniquely, with the focus of the ViT encoder on global relationships between frames and the ZigZag-enhanced AlexNet exploiting dilated convolutions for an extended receptive field. More robustness is provided by adding a DiMS-Inception network with multi-scale dilated convolutions, allowing the model to detect anomalies at different scales in video frames. The X2_PDDVnet model was thus presented to emerge with remarkable accuracy, precision, and sensitivity, especially achieving its peak accuracy of 0.9898 on the ShanghaiTech dataset. The integration of explainable AI techniques, such as Grad-CAM and LIME, into video anomaly detection, which helps explain visually why the model took its decision, is innovative in this work. Extensive testing on three datasets: UCSD, Avenue, and ShanghaiTech, showed X2_PDDVnet outperformed all baseline models on the different metrics, including F-measure, specificity, MCC, NPV, G-mean, and even with a lower FPR. The ablation study also shows the benefits of a dual-path architecture, because combined ViT and ZigZag encoders resulted in better performance metrics in all datasets, proving the viability of multi-encoder architectures for robust feature extraction. Furthermore, when tested with varied loss functions, the combined Focal and Dice Loss produced higher mIoU, mAP, and accuracy for segmentation tasks, hence demonstrating the importance of tailored loss functions for optimal model training. Future work could explore the expansion of this framework to various anomaly detection scenarios, further optimizing computational efficiency and adaptability to diverse environmental conditions.

ACKNOWLEDGMENTS

The authors would like to thank the Deanship of Karpagam Academy of Higher Education for supporting this work.

REFERENCES

[1] Kotkar, V.A., Sucharita, V. (2023). Fast anomaly detection in video surveillance system using robust spatiotemporal and deep learning methods. Multimedia Tools and Applications, 82(22): 34259-34286. https://doi.org/10.1007/s11042-023-14840-0

[2] Mangai, P., Geetha, M.K., Kumaravelan, G. (2024). Two-stream spatial-temporal feature extraction and classification model for anomaly event detection using hybrid deep learning architectures. International Journal of Image and Graphics, 24(6): 2450052. https://doi.org/10.1142/S0219467824500529

[3] Ganokratanaa, T., Aramvith, S., Sebe, N. (2022). Video anomaly detection using deep residual-spatiotemporal translation network. Pattern Recognition Letters, 155: 143-150. https://doi.org/10.1016/j.patrec.2021.11.001

[4] Habeb, M.H., Salama, M., Elrefaei, L.A. (2024). Enhancing video anomaly detection using a transformer spatiotemporal attention unsupervised framework for large datasets. Algorithms, 17(7): 286.

https://doi.org/10.3390/a17070286

[5] Aslam, N., Kolekar, M.H. (2022). Unsupervised anomalous event detection in videos using spatio-temporal inter-fused autoencoder. Multimedia Tools and Applications, 81(29): 42457-42482. https://doi.org/10.1007/s11042-022-13496-6

[6] Chen, A., Fu, Y., Zheng, X., Lu, G. (2022). An efficient network behavior anomaly detection using a hybrid DBN-LSTM network. Computers & Security, 114: 102600. https://doi.org/10.1016/j.cose.2021.102600

[7] Pavuluri, G., Annem, G. (2023). A Deep Learning Approach to Video Anomaly Detection using Convolutional Autoencoders. arXiv preprint arXiv:2311.04351. https://doi.org/10.48550/arXiv.2311.04351

[8] Shao, W., Rajapaksha, P., Wei, Y., Li, D., Crespi, N., Luo, Z. (2023). Covad: Content-oriented video anomaly detection using a self attention-based deep learning model. Virtual Reality & Intelligent Hardware, 5(1): 24-41. https://doi.org/10.1016/j.vrih.2022.06.001

[9] Shen, H., Shi, L., Xu, W., Cen, Y., Zhang, L., An, G. (2024). Patch spatio-temporal relation prediction for video anomaly detection. arXiv preprint arXiv:2403.19111. https://doi.org/10.48550/arXiv.2403.19111

[10] Abdullah, F., Abdelhaq, M., Alsaqour, R., Alatiyyah, M. H., Alnowaiser, K., Alotaibi, S.S., Park, J. (2023). Context aware crowd tracking and anomaly detection via deep learning and social force model. IEEE Access, 11: 75884-75898. https://doi.org/10.1109/ACCESS.2023.3293537

[11] Hwang, I.C., Kang, H.S. (2023). Anomaly detection based on a 3D convolutional neural network combining convolutional block attention module using merged frames. Sensors, 23(23): 9616. https://doi.org/10.3390/s23239616

[12] Abbas, Z.K., Al-Ani, A.A. (2022). Detection of anomalous events based on deep learning-BILSTM. Iraqi Journal of Information and Communication Technology, 5(3): 34-42.

[13] Qasim, M., Verdu, E. (2023). Video anomaly detection system using deep convolutional and recurrent models. Results in Engineering, 18: 101026. https://doi.org/10.1016/j.rineng.2023.101026

[14] Ul Amin, S., Ullah, M., Sajjad, M., Cheikh, F. A., Hijji, M., Hijji, A., Muhammad, K. (2022). EADN: An efficient deep learning model for anomaly detection in videos. Mathematics, 10(9): 1555. https://doi.org/10.3390/math10091555

[15] Kang, M., Lee, W., Hwang, K., Yoon, Y. (2022). Vision transformer for detecting critical situations and extracting functional scenario for automated vehicle safety assessment. Sustainability, 14(15): 9680. https://doi.org/10.3390/su14159680

[16] Kim, Y., Yu, J.Y., Lee, E., Kim, Y.G. (2022). Video anomaly detection using cross u-net and cascade sliding window. Journal of King Saud University-Computer and Information Sciences, 34(6): 3273-3284. https://doi.org/10.1016/j.jksuci.2022.04.011

[17] Le, V.T., Jin, H., Kim, Y.G. (2025). HSTforU: Anomaly detection in aerial and ground-based videos with hierarchical spatio-temporal transformer for U-net. Applied Intelligence, 55(4): 261. https://doi.org/10.1007/s10489-024-06042-4

[18] Balamurugan, G., Jayabharathy, J. (2022). An integrated framework for abnormal event detection and video summarization using deep learning. International Journal of Advanced Technology and Engineering Exploration, 9(95): 1494-1507. https://doi.org/10.19101/IJATEE.2021.875854

[19] Taghinezhad, N., Yazdi, M. (2023). A new unsupervised video anomaly detection using multi-scale feature memorization and multipath temporal information prediction. IEEE Access, 11: 9295-9310. https://doi.org/10.1109/ACCESS.2023.3237028

[20] Aslam, N., Kolekar, M.H. (2024). TransGANomaly: Transformer based generative adversarial network for video anomaly detection. Journal of Visual Communication and Image Representation, 100: 104108. https://doi.org/10.1016/j.jvcir.2024.104108

[21] Wu, C., Shao, S., Tunc, C., Satam, P., Hariri, S. (2022). An explainable and efficient deep learning framework for video anomaly detection. Cluster Computing, 25: 2715-2737. https://doi.org/10.1007/s10586-021-03439-5

[22] Qasim Gandapur, M., Verdú, E. (2023). ConvGRU-CNN: Spatiotemporal deep learning for real-world anomaly detection in video surveillance system. International Journal of Interactive Multimedia and Artificial Intelligence, 8(4): 88-95, https://doi.org/10.9781/ijimai.2023.05.006

[23] Sharif, M. H., Jiao, L., Omlin, C. W. (2023). CNN-ViT supported weakly-supervised video segment level anomaly detection. Sensors, 23(18): 7734. https://doi.org/10.3390/s23187734

[24] Yang, Y., Fu, Z., Naqvi, S.M. (2023). Abnormal event detection for video surveillance using an enhanced two-stream fusion method. Neurocomputing, 553: 126561. https://doi.org/10.1016/j.neucom.2023.126561

[25] Bajgoti, A., Gupta, R., Balaji, P., Dwivedi, R., Siwach, M., Gupta, D. (2023). Swinanomaly: Real-time video anomaly detection using video Swin transformer and SORT. IEEE Access, 11: 111093-111105. https://doi.org/10.1109/ACCESS.2023.3321801