Journal of Engineering Issues and Solutions

Unusual Activity Detection with Alert System Using Vision Transformer

Ocean Sitaula[1*], Prabesh Sharma[2], Pukar Karki[3], Shailesh Devkota[4], Suchita Kumari Sah[5], Pravin Sangroula[6]

[1,2,3,4,5] *Department of Electronics and Computer Engineering, Purwanchal Campus, Institute of Engineering, Tribhuvan University, Dharan, Nepal*

[6]*Department of Electronics and Computer Engineering, Pulchowk Campus, Institute of Engineering, Tribhuvan University, Kathmandu, Nepal*

**Corresponding email: ocinsitaula@gmail.com*

Submitted: April 13, 2025; Revised: June 15, 2025; Accepted: June 21, 2025

https://doi.org/10.3126/joeis.v4i1.81597

Abstract

Traditional surveillance systems are constrained by the limitations of manual monitoring, often resulting in delayed detection of anomalous activities. This research investigates an AI-driven surveillance system employing a Vision Transformer (ViT) architecture, pretrained on the UCF Crime Dataset, to automate anomaly detection by classifying video sequences as 'normal' or 'unusual'. The dataset was preprocessed by extracting frames at 30 fps, followed by resizing, augmentation, and fed into a fine-tuned ViT model, which achieved a macro F1-score of 0.7858 and 78.95% test accuracy. The system built in this study features a backend infrastructure based on Django, Django Channels, and Redis to enable efficient session management and WebSocket communication, supporting a live streaming module for anomaly detection and broadcasting via WebRTC, alongside a surveillance module for remote feed access. When any abnormal event is detected, the system automatically captures relevant snapshots of corresponding image and confidence score, and provides alert through the user interface or third-party applications. This approach aims to enhance detection accuracy, operational efficiency and situational awareness in surveillance environments.

Keywords: *Redis, Unusual Activity Detection, Vision Transformers (ViT), WebRTC, WebSocket*

1. Introduction

Ensuring the safety and security of public and private spaces has become increasingly reliant on intelligent surveillance systems, especially as urban environments grow more complex and densely populated (Tripathi et al., 2018). Traditional surveillance approaches, which depend on continuous human monitoring, often suffer from inefficiencies, fatigue, and the potential for missed incidents or delayed responses, particularly in high-activity or high-risk scenarios (Bhavyasri et al., 2023). The emergence of artificial intelligence (AI), deep learning, and advanced communication technologies such as WebRTC and WebSocket has revolutionized the field by enabling automated, real-time anomaly detection and rapid alerting mechanisms (Bhavyasri et al., 2023; Franklin et al., 2020). These advancements have led to the development of systems that can process live video streams, identify abnormal behaviors, and deliver immediate notifications to relevant authorities or stakeholders, significantly improving emergency response times and overall public safety outcomes.

Recent literature highlights the limitations of conventional deep learning models, particularly Convolutional Neural Networks (CNNs) and hybrid CNN-LSTM architectures, in the context of video surveillance anomaly detection (Hameed et al., 2024; M, T. & Singh, 2023; Pawar & Attar, 2019). While CNNs are effective in extracting spatial features from individual frames, they often struggle to capture long-range temporal dependencies that are crucial for understanding evolving events in video streams (Pawar & Attar, 2019). Hybrid approaches that combine CNNs with sequence models like LSTMs or GRUs have shown improved performance by analyzing frame sequences over time, yet they still face challenges in accurately detecting subtle or complex anomalies, especially in crowded or dynamic environments (M, T. & Singh, 2023). Moreover, these systems may generate false alarms or require substantial manual intervention for post-incident analysis, limiting their effectiveness for proactive threat prevention.

To address these challenges and identify the most effective solution for unusual activity detection, this research undertook a comprehensive exploration of multiple deep learning architectures and methodologies. The investigation began with traditional feature extraction approaches, progressing through various transformer-based solutions, ultimately converging on an optimized Vision Transformer implementation.

The first methodology employed VGG16 as a feature extractor combined with Long Short-Term Memory (LSTM) networks for temporal sequence modeling (M, T. & Singh, 2023). While VGG16 effectively captured spatial features from individual frames, the subsequent LSTM processing failed to achieve satisfactory performance in anomaly detection tasks. The limitations observed included insufficient temporal context understanding and suboptimal feature representation for complex unusual activities, resulting in poor classification accuracy and high false positive rates (Pawar & Attar, 2019).

Recognizing the potential of attention mechanisms for spatiotemporal analysis, the second approach involved developing a transformer architecture from scratch specifically tailored for video anomaly detection (Dosovitskiy et al., 2020). However, this implementation faced significant computational constraints during experimentation. The custom architecture, while theoretically sound, required substantial computational resources for training and inference, making it impractical for real-time surveillance applications given the available hardware limitations.

The third approach leveraged Google's ViViT-B-16x2-Kinetics400 model for direct video classification through fine-tuning. This method demonstrated superior performance compared to previous approaches, achieving better accuracy in unusual activity detection. However, this solution presented several limitations: reduced control over model outputs, inability to customize frame processing parameters, and computational constraints that prevented high-epoch training necessary for optimal performance. Additionally, the model's complexity hindered real-time inference capabilities essential for immediate alert generation.

Through iterative experimentation and analysis of the aforementioned approaches, the research identified Vision Transformers as the most promising solution when properly optimized for computational efficiency (Dosovitskiy et al., 2020; Hameed et al., 2024). The final methodology employs a Vision Transformer model fine-tuned on individual frames rather than entire video sequences, providing several key advantages: enhanced control over model outputs, customizable frame processing parameters for faster inference, and the ability to implement conditional alerting mechanisms when specific anomaly thresholds are met. Despite computational limitations, this approach enables effective real-time processing while maintaining high detection accuracy (Hameed et al., 2024; Joshi & Chaudhari, 2022).

Vision Transformers (ViTs), adapted from the field of natural language processing, have recently demonstrated superior performance in both spatial and temporal anomaly detection tasks within video surveillance applications. Leveraging self-attention mechanisms, ViTs can process entire video sequences simultaneously, capturing intricate spatiotemporal relationships and enabling more robust, interpretable, and accurate detection of unusual activities such as accidents, abuse, vandalism, or explosions. Comparative studies also indicate that Vision Transformers outperform traditional CNNs, LSTMs, and even CNN-LSTM hybrids across key benchmarks, including accuracy, precision, recall, and F1-score, particularly when applied to large-scale datasets like UCF Crime or XD-Violence (Hameed et al., 2024; Joshi & Chaudhari, 2022; Nazir et al., 2023). The integration of real-time communication protocols, such as WebRTC for low-latency video streaming and WebSocket for persistent, instant data exchange, further enhances the responsiveness and scalability of modern surveillance systems (Bhavyasri et al., 2023; Telikicherla et al., 2024).

This research, "Unusual Activity Detection with Alert System Using Vision Transformer," builds on this comprehensive theoretical and experimental foundation by developing an AI-powered surveillance solution that leverages a Vision Transformer model pretrained on the UCF Crime Dataset for binary classification of activities as normal or unusual. The system features live video streaming via WebRTC, real-time analysis using ViT, and instant alerting through secure third-party platforms such as Telegram, all orchestrated within a scalable backend architecture utilizing Django as the backend framework and Redis for session management and WebSocket channel layers.

The integration of Vision Transformer-based models with real-time communication technologies, informed by extensive experimental validation across multiple architectural approaches, represents a significant advancement in the field of intelligent surveillance. This research addresses critical gaps in existing solutions while demonstrating the importance of methodical approach evaluation in developing effective, scalable, and adaptive security frameworks.

2. Materials and Methods

2.1 Overview

The UCF Crime Dataset is a widely used benchmark for video-based anomaly detection in surveillance applications. It consists of 1,900 videos spanning 128 hours, covering 13 realistic anomalies such as arson, vandalism, fighting, and abuse. The dataset was converted into frames and divided into training and test subsets, with a total of 1,266,345 images for training and 111,308 images for testing. Each video's frames were extracted, resized to 64x64 pixels, and stored in PNG format for efficient processing. Preprocessing steps such as resizing and data augmentation, including rotation and sharpness adjustments, enhance model generalization were performed. The dataset was structured into "Normal" and "Unusual" classes and converted into a Hugging Face Dataset object for compatibility with the transformer library.

A pre-trained Vision Transformer (ViT) model was fine-tuned for binary classification (Normal or Unusual) using the 'google/vit-base-patch16-224-in21k' checkpoint. The training pipeline included hyperparameter tuning with learning rate, batch size, and number of epochs defined in Training Arguments. Model performance was evaluated using accuracy and F1-score, with a confusion matrix and classification report providing further insights. After training, the model was saved, and an inference pipeline was created for real-time anomaly detection in live footage.

This surveillance system integrates real-time video streaming, AI-based anomaly detection, and instant alerting system to enhance security monitoring. WebRTC enables low-latency live streaming, while WebSocket ensures seamless data exchange between the server and client devices. The AI model processes video frames, detects anomalies, and immediately triggers alerts. When an unusual activity is detected, the system captures a screenshot of the suspicious event and sends a real-time notification through third-party platforms like Telegram.

2.2 System Design

The system begins with capturing video from the live feed, which is processed in the backend. The processed frames are sent to the Vision Transformer model for anomaly detection. The model analyzes the frames and returns the results to the backend. The backend then displays the video feed and detection results on the surveillance interface. If an anomaly is detected, an alerting message is sent to a third-party application.

2.3 Data Collection

UCF Crime Dataset, a widely recognized benchmark for video-based anomaly detection in surveillance applications, was taken for the purpose of our research (Hameed et al., 2024; Joshi & Chaudhari, 2022; Nazir et al., 2023). The dataset comprises 1,900 videos with a total duration of 128 hours, covering 13 types of real-world anomalies. To facilitate training and testing, we converted the videos into frames, ensuring a structured representation of sequential information. The dataset was then systematically divided into training and test subsets:

- Training set: 1,266,345 frames
- Test set: 111,308 frames

2.4 Data Pre-Processing

The preprocessing pipeline involved organizing video frames into labeled categories, where normal activities were assigned a separate label from anomalous ones. Augmentation techniques such as resizing, random sharpness adjustments, normalization, and tensor conversion were applied to enhance model generalization. A pre-trained Vision Transformer (ViT) model was selected for feature extraction, and a processor was initialized to standardize input images to match the model's expected format. The dataset was then split into training and validation sets using a stratified approach, with numerical label mapping and a collate function to facilitate efficient batch processing during training.

2.4.1 Data Organization and Labeling

The extracted video frames were categorized based on their respective classes. The folder structure of the dataset was leveraged to assign labels to each frame. Normal activities were grouped under the "Normal" category, while all anomalous activities were collectively labeled as "Unusual". A structured dataset was created with each image associated with its respective label.

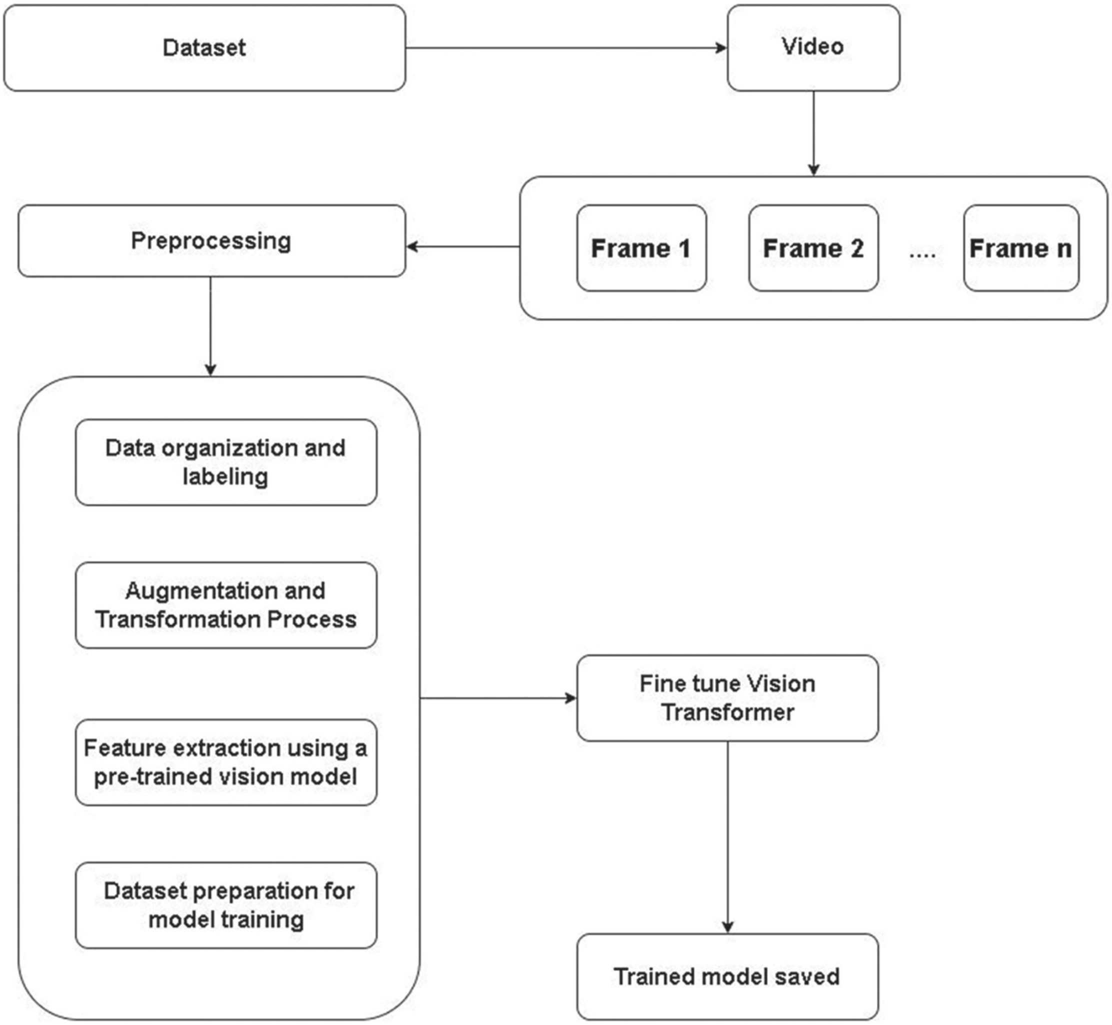
Dataset
Video
Frame 1
Frame 2
....
Frame n
Preprocessing
Data organization and labeling
Augmentation and Transformation Process
Feature extraction using a pre-trained vision model
Dataset preparation for model training
Fine tune Vision Transformer
Trained model saved

Figure 1: Data Pre-processing and Model Training

2.4.2 Augmentation and Transformation Process

To enhance the model's generalization capability and make it robust against variations, a series of transformations were applied to the training data:

- Resizing: Each image was resized to a fixed resolution to ensure uniform input size.
- Random Sharpness Adjustment: Some images were sharpened randomly to help the model recognize details better.
- Normalization: Pixel values were adjusted using the mean and standard deviation derived from a pre-trained vision model which ensured that input images are scaled correctly for optimal model performance.
- Conversion to Tensors: The images were transformed into tensor format for compatibility with deep learning frameworks. For validation and test data, only resizing, normalization, and tensor conversion were performed, avoiding random transformations to maintain consistency.

2.4.3 Feature Extraction Using a Pre-trained Vision Model

Feature extraction is a crucial step in utilizing deep learning models for image-based tasks, as it allows for the extraction of meaningful patterns from raw images. In this study, a pre-trained Vision Transformer (ViT) model was selected as the feature extractor. The ViT model is pre-trained on large-scale datasets, enabling it to capture intricate spatial relationships within images. Unlike traditional CNNs, which rely on convolutional operations, ViT processes images as sequences of patches, leveraging self-attention mechanisms to learn global and local dependencies (Dosovitskiy et al., 2020; M, T. & Singh, 2023; Nazir et al., 2023). This makes it highly effective for extracting features from complex datasets such as live footage. Before passing images into the ViT model, a processor was initialized to standardize the input format.

2.4.4 Dataset Preparation for Model Training

The dataset was split into training and validation sets using a stratified approach to maintain label balance. Each image label was mapped to a numerical ID to facilitate training. A collate function was defined to handle batch processing, ensuring efficient loading of image tensors and corresponding labels. This preprocessing pipeline ensured that the input data was structured, optimized, and ready for training in an anomaly detection framework.

2.5 Model Training

2.5.1 About the Model: Vision Transformer (ViT)

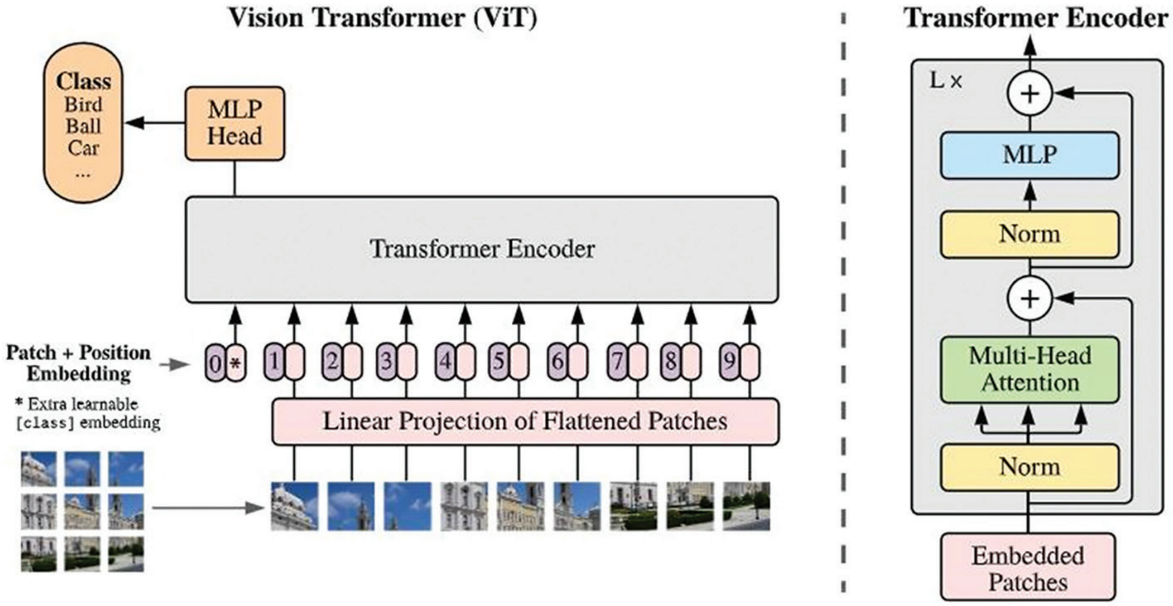

Figure 2: Architecture of Vision Transformer

The Vision Transformer (ViT) is a deep learning model that applies the transformer architecture, originally designed for natural language processing, to image classification tasks (Dosovitskiy et al., 2020; Hameed et al., 2024). Unlike traditional Convolutional Neural Networks (CNNs), which rely on convolutional layers to extract spatial features, ViT processes images as sequences of patches and applies self-attention mechanisms to learn relationships across the entire image. This global attention mechanism enables ViT to capture complex dependencies in images more effectively than CNNs, making it particularly suitable for anomaly detection in surveillance systems (Hameed et al., 2024). The model used in this study, ViT-Base (google/vit-base-patch16-224-in21k), is a pretrained version of ViT trained on ImageNet-21k, a large-scale dataset containing 14 million images across 21,843 classes. The model operates on 224x224 resolution images and divides each image into 16x16 pixel patches before processing them. Since ViT does not rely on convolutions, it is capable of learning fine-grained patterns across an image, which is beneficial for identifying unusual or suspicious activities in crime detection applications.

2.5.2 Model Initialization and Configuration

The pre-trained ViT model was loaded using the ViTForImageClassification class from the Hugging Face Transformers library. Since the base model was trained for generic image classification tasks, it needed to be fine-tuned to detect crime-related activities. To achieve this, a new classification head (a fully connected layer) was added to the model, adjusting the output layer to match the number of crime-related classes in the dataset. Additionally, label mappings were defined to associate numerical outputs with meaningful crime categories. Before feeding images into the model, they were resized to 224x224 pixels, ensuring compatibility with ViT's input format. Since the model was pre-trained with a specific data distribution, image normalization was applied using the same mean and standard deviation as the original training dataset. The ViT model subsequently transformed each image into a sequence of patch embeddings (16x16 pixel regions), which were passed through transformer layers. These embeddings allowed the model to analyze spatial relationships and patterns within the image effectively.

2.5.3 Training Configuration and Hyperparameters

Fine-tuning was performed using the Trainer API, which simplified model training and evaluation. The training process was configured with one epoch (which could be increased for better results), using a learning rate of 5e-6 to ensure stable updates to the model's weights. A batch size of 32 was set for training, while a smaller batch size of 8 was used for evaluation. To prevent overfitting, weight decay (0.02) was applied. Additionally, checkpoints were saved at the end of each epoch, allowing for model selection based on the best performance during training. After fine-tuning, the adapted ViT model was capable of identifying crime-related activities in surveillance images. By leveraging global attention mechanisms and feature extraction from patches, ViT provided a powerful solution for recognizing complex visual patterns in real-world anomaly detection scenarios.

2.6 Model Evaluation

Evaluating the trained model is a crucial step in determining its effectiveness in detecting unusual activities. Various metrics such as accuracy, loss, precision, recall, and F1-score are used to analyze the performance on validation and test datasets.

2.6.1 Performance on Validation Data

- Evaluation Loss: 0.0023
- Evaluation Accuracy: 99.94%

The evaluation on the validation dataset provided an initial assessment of the model's learning ability. The results indicated an extremely high accuracy of 99.94%, meaning the model effectively differentiated between normal and unusual activities within the training environment. The low loss value further confirmed that the model was able to extract meaningful patterns without significant errors.

2.6.2 Performance on Test Data

- Test Loss: 1.30
- Test Accuracy: 78.95%

Here, the test accuracy dropped to 78.95%, which is significantly lower than the validation accuracy. This suggests some degree of overfitting, meaning the model learned the training data too well but struggled slightly with generalizing to unseen test data.

2.6.3 Analysis of Predictions Using a Confusion Matrix

To gain deeper insights into model performance, a confusion matrix was used to evaluate its classification decisions. The model showed 79.01% accuracy in detecting crime-related frames and 78.91% accuracy in detecting normal activity frames. While these values appear balanced, the model demonstrated a slight bias toward normal activities, leading to some crime-related frames being misclassified as normal.

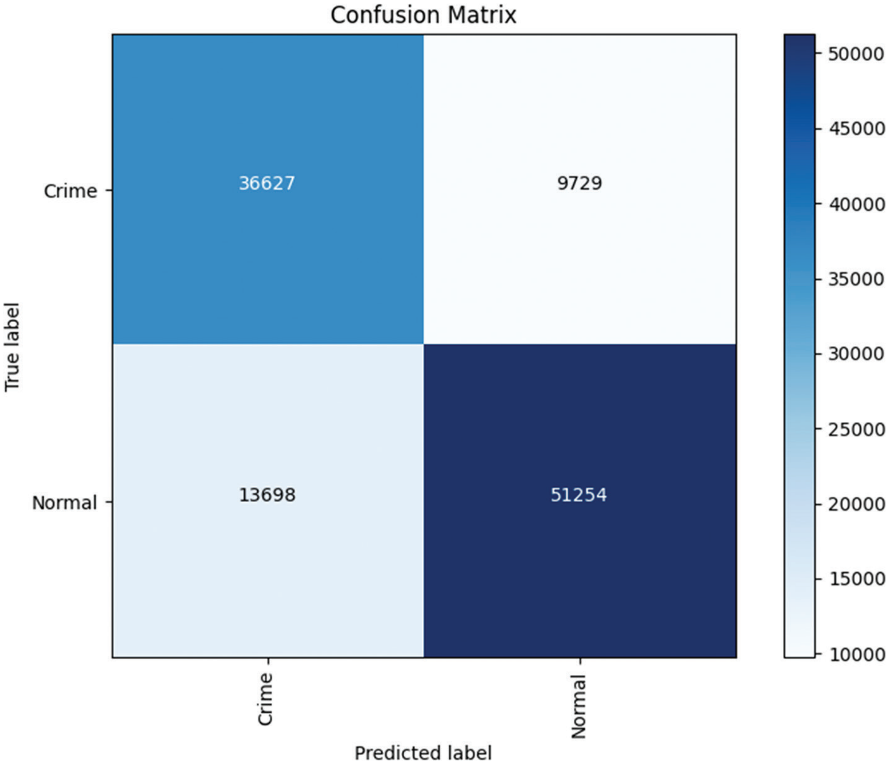

Figure 3: Confusion Matrix of Model

2.6.4 Classification Report

The classification report provided a more detailed breakdown of the model's predictive performance. Precision for crime detection was 72.78%, meaning that among the frames predicted as crime-related, about 72.78% were correct. However, the recall was 79.01%, indicating that the model correctly identified 79.01% of all actual crime-related frames.

For normal activity classification, the precision was relatively high at 84.05%, showing that most normal activity predictions were correct. However, recall was 78.91%, meaning that some normal frames were mistakenly identified as crime-related. The overall macro F1-score of 0.7858 suggests that while the model performed fairly well, there is still room for optimization to improve its generalizability and accuracy.

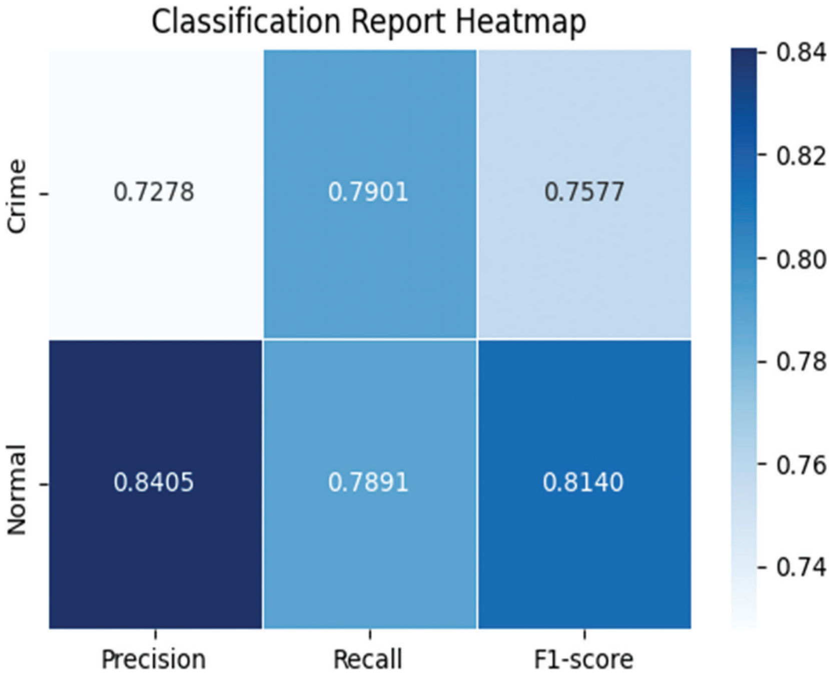

Figure 4: Classification Report of Model

2.6.5 Justification of model

The model shows strong validation performance but struggles to generalize, with test accuracy dropping to 78.95%, indicating overfitting. A major factor is resource limitations, as training on large video frames took over 11 hours for just one epoch, restricting the model's ability to refine its parameters. The complexity of video-based anomaly detection further adds to the challenge, as crime-related activities are highly context-dependent. The confusion matrix shows that the model sometimes misclassifies events, with a tendency to either label normal activities as crimes (false alarms) or fail to detect actual crimes (missed detections). This could be due to the structure of the crime datasets, which include portions of videos and abundant frames that depict normal activities, making it harder for the model to differentiate between normal and suspicious behavior. Despite these constraints, the model achieves a macro F1-score of 0.7858, showing it can effectively detect unusual activities but is impacted by computational constraints and limited training iterations.

2.7 Session Management

Session management is a critical component of the system, facilitating separate sessions for each live stream, allowing multiple video feeds to be managed concurrently without interference. It ensures that only authorized users can access their designated streams, maintaining secure and organized control.

The system is designed to handle session creation, retrieval, tracking, and termination seamlessly by using Redis as database for session data storage. The Live Stream module is responsible for initiating a live session. When a new live stream session begins, an OTP is generated along with a unique session ID. The OTP serves as an access key, ensuring that only authorized users can join the stream. This information is stored in Redis, which enables quick retrieval of active sessions. The session remains active until it is manually terminated.

On the Surveillance module, users must enter the correct OTP to find and connect to an active session. The system verifies the OTP from stored session data, retrieving the corresponding session ID. This structured authentication prevents unauthorized access and ensures that only users with valid OTPs can view the live footage. When a stream ends, the session must be properly terminated. The system handles this by deleting session data from Redis and invalidating the OTP which optimizes resource usage by eliminating inactive sessions.

Thus, session management system ensures seamless real-time communication, secure access control, and efficient cleanup of inactive sessions, contributing to a well-structured and optimized live streaming experience.

2.8 Real-Time Communication

Real-time communication is a key aspect of the system, enabling seamless live video streaming and surveillance functionalities. The architecture relies on Django Channels, WebSockets, WebRTC, Daphne, and Redis to ensure smooth and low-latency transmission of video feeds. These technologies work together to establish, manage, and maintain connections between live streamers and surveillance viewers. The system is designed to handle multiple concurrent sessions efficiently while ensuring reliable communication and scalability.

The system uses WebSocket for signaling by exchanging connection details such as session IDs, offers, answers, and ICE candidates between streamers and viewers. Django Channels manages these WebSocket connections through consumer classes, grouping users into Redis-backed Django groups based on their session. Each stream is associated with a dedicated group, where both the streamer and connected viewers are dynamically added. The streaming module sends signaling data to this group via WebSocket, ensuring all viewers receive real-time connection updates necessary to establish a stable WebRTC connection.

The Live Stream module initiates the real-time transmission process. When a user starts streaming, the browser's 'getUserMedia' API captures the video feed from the user's camera. This video and audio data is then transmitted using WebRTC, which enables peer-to-peer communication for smooth and low-latency streaming by allowing direct communication between devices without requiring an intermediary server.

On the Surveillance module, users can enter the OTP to access an active stream. Once authenticated, the WebSocket connection retrieves the session information, and WebRTC is used to establish a direct video stream between the peers. Django groups facilitate this by ensuring that each connected surveillance user receives updates and signaling messages related to their session. Unlike traditional HTTP-based streaming, which introduces significant latency, the combination of WebRTC, WebSocket, and Django groups allows for near-instantaneous video communication, making the system ideal for real-time surveillance applications.

WebSocket facilitates the signaling process required to establish a WebRTC connection by managing essential interactions such as connection initiation, offer-answer exchange, and ICE candidate negotiation. When a user starts a live stream, the browser captures media from the camera and microphone using the getUserMedia API, generating an 'offer' that includes details about the media stream and connection parameters. This 'offer' is transmitted via WebSocket to the Surveillance module, where a user attempting to join responds with an 'answer', completing the handshake and establishing the connection.

For direct peer-to-peer communication to be established, both the streamer and viewer must determine a viable network path. This is achieved through the exchange of ICE (Interactive Connectivity Establishment) candidates, which contain network-related data such as IP addresses and ports. These candidates, transmitted over WebSocket, allow peers to establish a stable communication route, even in complex network environments involving NATs or firewalls. Once a viable route is determined, media is streamed directly between the streamer and viewer using WebRTC, bypassing any intermediary server to ensure low-latency, high-quality video transmission.

Although media exchange is handled by WebRTC after the connection is established, WebSocket remains active to manage the session. It oversees viewer connections, detects disruptions, supports session termination, and facilitates the renegotiation of ICE candidates when needed—especially under changing network conditions. WebRTC also adapts video quality dynamically based on bandwidth and latency. In cases where multiple viewers connect simultaneously, WebRTC efficiently handles multiple peer connections. When a session concludes, WebSocket notifies all participants, closes connections, and releases associated resources, ensuring stable and efficient lifecycle management of the live stream.

To handle multiple concurrent streams efficiently, Redis is used instead of the default in-memory channel layer. Redis acts as a message broker, routing and storing WebSocket messages between consumers. It also manages session storage, ensuring that multiple users can access the same stream. Unlike in-memory storage, which is limited to a single server instance, Redis provides persistence and scalability, allowing the system to support a large number of live streams and surveillance connections simultaneously.

Daphne serves as the ASGI server, handling WebSocket connections and managing real-time communication. It works alongside Django Channels to ensure that WebSocket requests are processed efficiently. Since real-time communication requires asynchronous processing, Django's traditional synchronous request response cycle, implemented in WSGI server, is bypassed in favor of event-driven WebSocket communication, which Daphne and Django Channels facilitate.

By integrating these technologies, the system provides a scalable, high-performance real-time video streaming solution. Users can start live sessions, connect to streams via the Surveillance module, and receive instant video feeds with minimal latency.

2.9 Model Integration in Backend

The trained AI model is seamlessly integrated into the backend, where it operates directly on the video stream generated on the Live Stream page. The video feed is captured every 5 seconds and processed into individual frames at a rate of 30 frames per second (FPS).

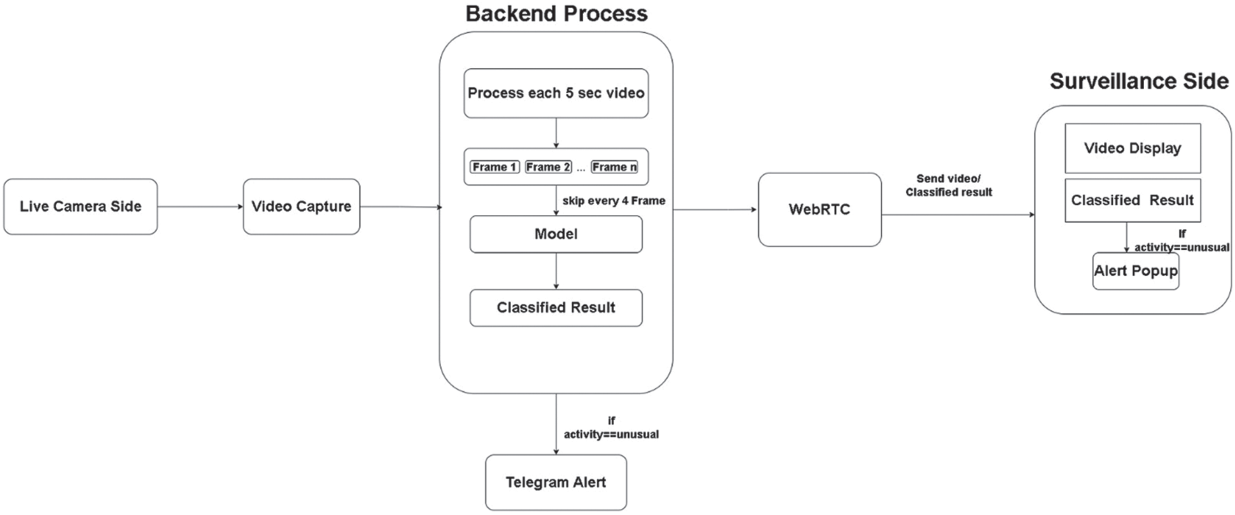

Figure 5: Backend Process with Model

The model was designed to analyze every 5th frame while skipping the preceding 4 frames to ensure real-time analysis without overloading system resources. This approach significantly optimizes performance while maintaining high detection accuracy. During inference, the model scans the selected frames for anomalies, identifying potential threats or unusual activities. Since our model is overfitted, we set a threshold: if an anomaly is detected in two frames within a 5-second interval, the system triggers an alert. This ensures that the alert system activates only when abnormal activities are consistently detected. The alert can be transmitted via WebSocket or external notification channels, enabling continuous monitoring and proactive security responses without disrupting the live streaming experience.

2.10 Automated Alerting System

The system integrates an automated alerting mechanism that notifies users upon detecting suspicious activity (Bhavyasri et al., 2023; Tripathi et al., 2018). This has been achieved through two types of alerts: alerts through third-party application and pop-up on the surveillance interface. These alerts ensure that security personnel remain informed, even if the surveillance interface is not actively monitored.

2.10.1 Background Third-Party Application Alerts

The live stream undergoes continuous AI-based monitoring and when an anomaly is detected, the system identifies the frame with the highest abnormality score and sends it to an external application's bot, accompanied by an alerting message. This ensures that security personnel receive timely alerts and can take immediate action, even when the surveillance page is not being monitored.

2.10.2 Real-Time Surveillance Interface Pop-Up

When a personnel is continuously monitoring the live footages a popup alert appears in the surveillance interface if any abnormal activity is detected. The system displays additional information like confidence score and captured frame number of the suspicious activity which is being sent to external application bot. Additionally, an alert sound is triggered to ensure immediate attention, in case the pop-up goes unnoticed.

3. Results and Discussion

3.1 Model Performance on Validation and Test Data

The model achieved an evaluation loss of 0.0023 and an evaluation accuracy of 99.94%. This demonstrates extremely high accuracy on the validation set, confirming the model's ability to effectively distinguish between normal and abnormal activities within the training environment.

The model exhibited a test loss of 1.30 and a test accuracy of 78.95%. The significant drop in accuracy on the unseen test set highlights a gap between validation and test performance. This indicates overfitting a common challenge in deep learning, particularly with complex video datasets. The high validation accuracy suggests the Vision Transformer (ViT) model learned training data patterns robustly, but its sensitivity to dataset diversity and training regimes (Dosovitskiy et al., 2021) limited generalization to new data.

3.2 Detailed Analysis of Model Predictions

The confusion matrix shows a crime-related frame detection accuracy of 79.01% and a normal activity detection accuracy of 78.91%, with a slight bias toward normal activity. The classification report indicates a precision of 72.78% and recall of 79.01% for crime detection, while normal activity achieved 84.05% precision and 78.91% recall. The overall macro F1-score is 0.78581, reflecting balanced performance across both classes.

These results demonstrate the model's robustness in handling surveillance data. The higher recall for crime detection suggests it effectively flags potential threats, which is critical in minimizing missed incidents. However, the lower precision indicates a tendency for false positives—a known challenge in anomaly detection systems highlighting a trade-off between sensitivity and specificity.

3.3 Justification and Limitations of the Model

While the macro F1-score of 0.7858 reflects effective detection performance, generalizability remains limited. Training was constrained by resource limitations, with each epoch requiring over 11 hours, restricting optimization and contributing to overfitting. Additionally, the presence of normal frames within crime-labeled videos complicated learning and introduced label noise a recognized challenge in surveillance datasets. These findings are consistent with literature indicating that ViTs benefit from extensive training and balanced data.

The model was deployed in a real-time surveillance system, analyzing every 5th frame to balance performance and resource usage. Alerts were triggered when anomalies appeared in at least two frames within a 5-second window, helping reduce false positives. Notifications were sent via third-party applications and real-time pop-ups, including confidence scores and frame details.

This real-time integration demonstrated the system's practical utility by supporting immediate response and continuous monitoring (Bhavyasri et al., 2023). The dual alerting mechanism ensured event detection even when the interface was unattended, while the multiple-frame threshold aligned with best practices in reducing noise. Performance results were consistent with recent studies showing that ViTs outperform CNN and hybrid models in video anomaly detection, particularly on large-scale datasets like UCF Crime (Hameed et al., 2024). Despite this, challenges such as overfitting and label ambiguity remain and point to future work in data balancing, longer training cycles, and further model optimization.

4. Conclusions

The system successfully demonstrates an AI-powered surveillance solution capable of real-time anomaly detection and alerting by leveraging a Vision Transformer (ViT) model trained using the UCF Crime Dataset for video analysis, alongside a third-party application API for alerting. The system effectively enhances security monitoring with minimal human intervention and addresses limitations like human fatigue, slow response times, and limited scalability in traditional surveillance systems. The system can detect suspicious activities such as accidents, abuse, violence, vandalism, explosions, etc. in real-time by classifying frames as normal or unusual. Upon detection, the system immediately captures snapshots of frames and transmits alerts to concerned authorities ensuring rapid responses to potential threats. Collectively, this research not only automates security monitoring, but also highlights the transformative potential of Vision Transformer architectures in modelling comprehensive spatial information which helps it to excel in numerous domains reliant on visual analysis.

Acknowledgements

We would like to express our sincere gratitude to the Department of Electronics and Computer Engineering, Purwanchal Campus, Dharan for providing the academic environment, guidance, and resources necessary to conduct this research. We acknowledge the use of the UCF Crime Dataset, which served as the primary data source for model training and evaluation. We are also grateful for the technical and administrative support extended by our teachers, faculty members, and colleagues, whose contributions were invaluable to the successful completion of this research.

References

Bhavyasri, J., Ramaiah, D. G. N. K., & Rasadurai, D. K. (2023). AI based smart surveillance system. International Journal of Scientific Research in Science, Engineering and Technology. https://doi.org/10.32628/IJSRSET229672

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. https://doi.org/10.48550/arXiv.2010.11929

Franklin, R. J., Mohana, & Dabbagol, V. (2020). Anomaly detection in videos for video surveillance applications using neural networks. Proceedings of the 4th International Conference on Inventive Systems and Control, ICISC 2020. https://doi.org/10.1109/ICISC47916.2020.9171212

Hameed, S., Amin, J., Anjum, M., & Sharif, M. (2024). Suspicious activities detection using spatial–temporal features based on vision transformer and recurrent neural network. Journal of Ambient Intelligence and Humanized Computing, 15. https://doi.org/10.1007/s12652-024-04818-7

Joshi, M., & Chaudhari, J. (2022). Anomaly detection in video surveillance using slowfast resnet-50. International Journal of Advanced Computer Science and Applications, 13. https://doi.org/10.14569/IJACSA.2022.01310112

M, T., & Singh, J. (2023). Unusual crowd activity detection in video using CNN, LSTM and OpenCV. International Journal for Research in Applied Science and Engineering Technology, 11. https://doi.org/10.22214/ijraset.2023.55240

Naik, B. T., Hashmi, M. F., & Bokde, N. D. (2022). A comprehensive review of computer vision in sports: Open issues, future trends and research directions. Applied Sciences, 12. https://doi.org/10.3390/app12094429

Nazir, A., Mitra, R., Sulieman, H., & Kamalov, F. (2023). Suspicious behavior detection with temporal feature extraction and time-series classification for shoplifting crime prevention. Sensors, 23. https://doi.org/10.3390/s23135811

Pawar, K., & Attar, V. (2019). Deep learning approaches for video-based anomalous activity detection. World Wide Web, 22. https://doi.org/10.1007/s11280-018-0582-1

Telikicherla, L. S., Sai, C. U., Chandini, A., & Dhanalaxmi, C. (2024). Deep learning enhanced human activity recognition and behaviour analysis with OpenCV integration (pp. 1–5). https://doi.org/10.1109/CINE63708.2024.10882065

Tripathi, R. K., Jalal, A. S., & Agrawal, S. C. (2018). Suspicious human activity recognition: A review. Artificial Intelligence Review, 50. https://doi.org/10.1007/s10462-017-9545-7