# Detection of Video Anomalies via CNN-LSTM Model for Intelligent Surveillance

Mohamed H. Mousa[1], Yasser M. Ayid[2], Ayman E. Khedr[3], Ahmed M. Elshewey[4]
Department of Computer Science and Artificial Intelligence, College of Computer Science and Engineering, University of Jeddah, Jeddah 21493, Saudi Arabia[1]
Mathematics & Statistics Department, College of Science, University of Jeddah, Jeddah, Saudi Arabia[2]
Department of Information Systems, College of Computing and Information Technology at Khulais, University of Jeddah, Jeddah 21959, Saudi Arabia[3]
Faculty of Computers and Information-Department of Computer Science, Suez University, P. O. Box: 43221, Suez, Egypt[4]

***Abstract*—Automated Video Anomaly Detection (VAD) plays a vital role in developing surveillance systems in public spots. Our study develops real-time anomaly detection via a hybrid Convolutional Neural Network–Long Short-Term Memory (CNN-LSTM) model, which uses the UCSD Pedestrian (Ped2) dataset. It introduces a methodology designed for detection accuracy enhancements by extracting CNN-based spatial features combined with learning LSTM-based temporal sequences. Preprocessing manages the class imbalance issue throughout several phases, including frame extraction, resizing, normalization, augmentation, and SMOTE balancing. Regarding the evaluation phase, several metrics such as accuracy, precision, recall, F1-score, and AUC are applied, indicating the superior performance of the CNN-LSTM model, which could outperform both the standalone CNN and LSTM models, having 93.5% accuracy, 91.8% precision, 90.2% recall, 91.0% F1-score, and an AUC of 0.947. Conclusively, our methodology is designed for improving the accuracy of the detection phase by integrating CNN-based spatial feature extraction along with LSTM-based temporal sequence learning.**

***Keywords*—Video Anomaly Detection; smart surveillance; computer vision; CNN-LSTM; hybrid deep learning**

## I. Introduction

The manual detection of video streams is gradually getting impractical due to the rapid expansion of surveillance systems in public, private, and critical infrastructure spots. In broader contexts such as smart homes, anomaly detection systems have also gained significant traction using machine learning approaches [1]. Human operators are prone to subjectivity, distraction, and fatigue, which may lead to missed important events or delayed responses. This means automated, intelligent video surveillance systems that can perform precise, real-time detection of unusual or suspicious activities are increasingly demanded.

In the computer vision and artificial intelligence domains, VAD is dedicated to detecting events in a video stream that deviate from typical behavioral patterns. These anomalies may encompass a broad spectrum of events, including theft, vandalism, vehicle intrusion into pedestrian areas, or unusual crowd behavior. The aim is to automatically flag such events for human review or automated response, thereby improving operational effectiveness and public safety.

The traditional VAD systems have demonstrated inefficacy compared to the rule-based algorithms and handcrafted elements, which can outperform these traditional ones. Those systems lack adaptability and generalization in dynamic or complex real-world settings, but they perform effectively in controlled ones. The latest enhancements in deep learning have significantly altered VAD by managing to select the most important features automatically from raw video data, emphasizing the shortcomings of manual feature engineering.

RNNs and CNNs deal with video sequences in a different manner. While RNNs, particularly LSTMs, have outperformed in capturing temporal dependencies from video sequences, CNNs have demonstrated superior performance in capturing spatial features from individual shots. Several promising hybrid models provide more precise anomaly detection by integrating CNN along with LSTM for better access to both spatial and temporal features.

Despite rapid progress in VAD, four practical gaps persist on standard benchmarks such as UCSD Ped2: cross-paper protocol inconsistency that prevents comparisons; lack of a systematic comparison between data-level (e.g., SMOTE/augmentation) and loss-level (e.g., class-balanced/focal) imbalance remedies under the same training setup; limited evidence on how far a lightweight spatiotemporal hybrid can go for real-time deployment relative to heavier models; and insufficient attribution of gains via ablations, confidence intervals, and statistical tests.

To address the gap, our study performed a controlled evaluation of CNN, LSTM, and CNN-LSTM baselines, providing a unified and reproducible pipeline. Additionally, we focused on examining imbalance strategies and conducting compute-aware analysis. This approach is not just a new architecture but is considered a transparent, deployment-oriented baseline working under a single protocol to quantify the incremental value of temporal modeling and rare-event remedies.

This study is considered a contribution to intelligent video surveillance, developing and evaluating a deep learning-based system intended for Video Anomaly Detection. It mainly integrates LSTM along with CNN in a hybrid deep learning model for VAD, upscaling detection accuracy and robustness. It gathers realistic surveillance scenarios along with minor

anomalies in the UCSD Pedestrian dataset for the purpose of more precise evaluation. Regarding the class imbalance issue, it applies SMOTE and augmentation approaches to a complete data pipeline, introducing a reproducible study.

The research develops simple models that could manage the complexity issue, demonstrating computational efficiency and flexibility in handling dynamic settings. Our study indicates the efficacy of the CNN-LSTM hybrid model in anomaly detection when drawing a comparison between CNN, LSTM, and CNN-LSTM models, which could surpass other models and achieve 93.5% accuracy and an AUC of 0.947. When evaluating anomaly detection strategies and comparing detection performance and computational cost, we could find out several strategies for selecting models in real-time or resource-constrained VAD applications.

Ultimately, this study provides a foundation for future studies and practical implementation by suggesting improvements, including real-time deployment via model compression techniques, cross-dataset generalization via domain adaptation, and improved detection reliability through multimodal fusion of video and audio.

Amin et al. [2] managed to partition the input video by deploying a shot boundary detection algorithm to an effective Anomaly Detection Network (EADN), along with a simple deep learning model. The model is designed to select spatiotemporal features via a time-distributed CNN as well as capture temporal dependencies across LSTM layers, providing strong real-time detection on benchmark datasets.

Hao et al. [3] attempted to address the temporal inconsistency and ambiguous anomaly boundaries via a spatiotemporal consistency-enhanced network. To generate synthetic frames, they deployed a 3D CNN-based encoder along with a 2D decoder, while the discriminator is applied to evaluate spatiotemporal consistency. Conclusively, their methodology is designed for better coherence in frame predictions, improving anomaly detection.

Qiu et al. [4] managed to design a dual-scale feature-clustering module as a convolution-enhanced self-attentive video autoencoder based on a U-Net architecture. To differentiate between both normal and abnormal events more effectively, they applied this module, which could integrate both spatial and channel-wise features. They designed another scoring methodology to mitigate temporal leakage, handling intricate datasets more effectively.

Huang et al. [5] discussed the overfitting limitations of deep generative models and how to address them via a Self-Supervised Attentive Generative Adversarial Network (SSAGAN). They integrated a self-attentive predictor along with dual discriminators (vanilla and self-supervised via rotation detection) into this network for improved generalization and anomaly detection using semantic prediction errors.

Shin et al. [6] integrated a CLIP-based semantic module along with an I3D motion sequence and came up with a weakly supervised mechanism: WS-VAD. They utilized a graph-based model, incorporating GCNs along with a multi-head attention mechanism for feature splice. When evaluating the model on UCF-Crime and XD-Violence datasets, they observed its effectiveness in examining untrimmed videos.

Mishra and Jabin [7] developed deep autoencoders to support unsupervised identification of spatiotemporal features from surveillance cameras. They managed to identify anomalous behaviors across the UCSD Ped1 and CUHK Avenue datasets via a regularity-score-based thresholding mechanism, coming up with competitive AUC scores.

Li and Tong [8] proposed a robust 3D autoencoder framework enhanced with multi-level feature splicing and joint multi-task learning. They extracted two types of features: spatial features, which involve noise replications, and temporal features, which involve frame reversal and deletion for mitigating abnormal replications, and they succeeded in improving the feature extraction process via attention gating modules. Their model came up with superior AUC scores within the UCSD Ped2 and CUHK Avenue datasets.

Additionally, advancements in adaptive object tracking, such as the Deep SORT enhancements in [9], further complement anomaly detection by maintaining precise localization and identity tracking in crowded scenes.

These studies emphasize the effectiveness of hybrid models that utilize spatial and temporal cues, unlike those single-stream approaches that cannot act with spatial and temporal features. Nevertheless, class imbalance, high false positive rates, and deployment feasibility on resource-constrained edge devices still form significant issues. Accordingly, 1) we draw a detailed comparison between CNN, LSTM, and CNN-LSTM and discuss a reproducible VAD pipeline, attempting to address confounding and quantify the added value of temporal modeling. 2) We analyze class-imbalance strategies (SMOTE vs. class-balanced/focal losses) and report their impact on Ped2. 3) We provide deployment-aware design choices (compact backbone, sequence windowing) and a clear path to model compression for real-time use. 4) We conduct an enhanced comparison on UCSD Ped2 along with unified evaluation (frame-level AUC) and statistical testing, emphasizing the competitiveness of our hybrid model while being simple and practical.

## II. Methodology

### A. Dataset

We evaluate the VAD mechanism, leveraging the well-known UCSD Pedestrian dataset [10] within the VAD research community. This dataset is well-suited for assessing the generalizability and robustness of different anomaly detection models, as it is designed to detect minor, context-sensitive abnormalities in crowded pedestrian environments.

The UCSD Pedestrian dataset is partitioned into a couple of distinct subsets—Ped1 and Ped2—captured from several camera angles on a university campus. Our study is focused on the UCSD Ped2 subset due to its controllable complexity and distinct annotations of anomalies. The dataset gathers normal events and anomalous events as two main types of recordings. The normal events are all of pedestrians walking along approved paths, while the anomalous ones involve cyclists, cars, or people walking in restricted areas as non-pedestrians.

The dataset videos are shot at 240x360 pixels each, having a frame rate of roughly 10 frames/second. Two main components of the dataset are the training set, which consists of just normal videos. This data is absolutely important for weakly supervised and unsupervised learning.

### B. Data Preparation

There are several steps in both data training and evaluation processes, starting with frame extraction and then partitioning each video into single frames. Subsequently, we resize frames to 224×224 pixels to meet all the requirements of traditional deep learning models.

We evaluate three remedies for rare anomalies under identical backbones and schedules: 1) SMOTE applied on frame-level minority samples; 2) class-balanced loss, which re-weights by the effective number of samples; and 3) focal loss ($\gamma$ tuned on validation). SMOTE was selected for the mainline because it preserves the cross-entropy objective and can improve minority recall on small datasets; the loss-level methods provide an alternative when synthetic sampling is undesirable.

Normalization was conducted after the resizing process, where the pixels were scaled to the [0, 1] range. We notice that the application of data augmentation approaches, including horizontal flipping and random rotation, can support the dataset variability and mitigate the overfitting risk.

The images in every frame are resized to 224x224 pixels and are processed in a series of 10 consecutive frames at once. This strategy represents the balancing point of various factors such as spatial resolution, temporal window size, and the need for efficient computation and stable training. Although increasing the resolution and the length of the temporal window would lead to better results, it also adds computational cost and makes the task harder.

### C. The Proposed Methodology

This study supports real-time VAD (Video Anomaly Detection) across the UCSD Pedestrian dataset via a deep learning-based methodology. This methodology presents three models: CNN, LSTM, and a hybrid CNN-LSTM for both spatial and temporal feature extraction. It is especially designed for normal behavior identification in videos throughout the training phase, while detecting anomalies throughout testing.

This methodology involves thorough data preprocessing as its first phase, in which the UCSD Ped2 dataset is partitioned into individual sequences and then resized to 224×224 pixels each in the models for unified input dimensions. For stabilized learning, these pixel values are normalized to the [0, 1] range, and for enhanced generalization to unseen data, the model employs random horizontal flipping, rotation, and brightness adjustment as effective data augmentation techniques. When training models, the VAD datasets with inherent class imbalance—where the normal events are much more common than abnormal ones—utilize SMOTE for further synthetic anomaly samples and then balance the dataset and mitigate the model bias.

This study addresses three deep learning models. The first is a CNN, which particularly captures spatial features from individual video sequences. This model learns static visual patterns across dense layers, which follow several other convolutional and pooling layers. The second one is an LSTM network. Unlike CNN, it can capture temporal features from video sequences, take its input features from a frame-level sequence, and identify movement direction or prolonged loitering as time-based behavioral patterns. The third model is a hybrid, integrating CNN along with LSTM (CNN-LSTM model). It demonstrates high effectiveness in processing video sequences, first by a CNN to capture spatial statistics, and then the resulting features are transferred to an LSTM to extract temporal dynamics. This means the model can detect both appearance-based and behavior-based anomalies accurately.

In the anomaly detection phase, our models are trained on normal video feeds via common learning paradigms (unsupervised or weakly supervised). The models learn how to identify normal patterns throughout their training; as a result, any significant deviation means serious prediction errors. Utilizing the Adam optimizer, the learning rate and batch size as hyperparameters are carefully selected, thereby enhancing the training phase. The reconstruction-based models with classification layers utilize categorical cross-entropy; otherwise, MSE is considered their primary loss function. A part of the dataset gathered in the training phase is utilized as a guarantee for effective generalization as well as to prevent overfitting.

Anomalies are detected during the testing phase by calculating an anomaly score for each frame or sequence, which is typically derived from the reconstruction error or confidence levels in the classification output. Throughout the validation phase, a threshold is utilized for frame classification to be normal or anomalous, making the final decision for detection at the frame level; this supports real-time or near-real-time anomaly alarms in surveillance scenarios.

The CNN backbone utilized in this study comprises an efficient custom-designed model that enables the network to perform effective spatial feature extraction from video frames. To that end, the CNN model comprises three convolutional blocks, wherein each of the blocks contains a two-dimensional convolutional layer (Conv2D) with filter size (3×3), followed by a Rectified Linear Unit (ReLU), batch normalization, and max-pooling with pool size (2×2). The number of convolutional layers is set to 32, 64, and 128.

Finally, the CNN feature map outputs are flattened into a one-dimensional vector and subsequently reshaped into sequences in the temporal dimension and fed to the LSTM layer for learning temporal features. In addition, dropout with a ratio of 0.3 is added following the convolutional blocks in order to avoid overfitting.

The presented models are thoroughly evaluated in accordance with multiple standard classification metrics: Accuracy, Precision, Recall, F1 Score, and AUC; these metrics assess the model's effectiveness in detecting both true anomalies and normal events. When observing Fig. 1, we find it depicts the workflow of the methodology proposed for VAD, whereas Algorithm 1 illustrates the CNN-LSTM for VAD.

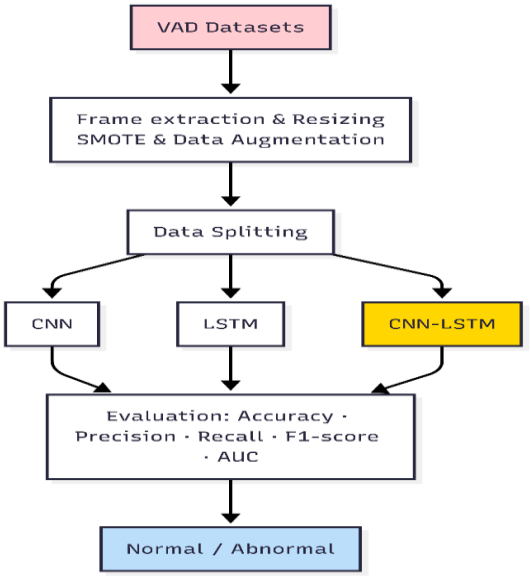


Fig. 1. Workflow diagram of the proposed VAD methodology.

**Algorithm 1:** CNN-LSTM for Video Anomaly Detection

**Input: VAD Video Dataset D**

**Output: Anomaly classification (Normal / Abnormal) for each frame**

1. BEGIN
2. // Step 1: Data Preprocessing
3. FOR each video V in Dataset D DO
4. Extract frames from V
5. Resize each frame to 224x224 pixels
6. Normalize pixel values to [0, 1]
7. Apply data augmentation (e.g., flipping, rotation)
8. END FOR
9. Apply SMOTE to balance class distribution
10. Split the preprocessed data into:

- Training set (70%), Validation set (15%) and Testing set (15%)

11. // Step 2: CNN-LSTM Model Architecture
12. Define CNN layers for spatial feature extraction:

- Conv2D → ReLU → MaxPooling → BatchNorm (repeat as needed)

13. Flatten CNN output to a 1D feature vector
14. Reshape feature vector into a time-series sequence
15. Pass sequences to LSTM layers:

- LSTM → Dropout → LSTM → Dense

16. Add output layer:

- Dense → Sigmoid (for binary classification)

17. Compile the model with:

- Optimizer: Adam
- Metrics: Accuracy, Precision, Recall, AUC

18. // Step 3: Model Training
19. Train CNN-LSTM model on training set
20. Validate using validation set
21. Apply early stopping and checkpoint saving if needed
22. // Step 4: Anomaly Detection & Evaluation
23. FOR each frame in testing set DO
24. Predict anomaly score using trained CNN-LSTM
25. IF anomaly score > threshold THEN
26. Label = "Abnormal"
27. ELSE
28. Label = "Normal"
29. END IF
30. END FOR
31. Assess model using:

- Accuracy, Precision, Recall, F1-Score, AUC

32. RETURN classification labels and evaluation metrics
33. END

### *D. Experimental Setup*

When implementing the models, we developed multiple common deep learning mechanisms: TensorFlow and PyTorch, as well as Python 3.8. As a result of the experiments conducted on a high-performance machine prepared with an NVIDIA RTX 3090 GPU, an Intel Core i9 CPU, 64 GB of RAM, and a 1 TB SSD, we observed rapid improvements in both training and evaluation.

Each model is comprehensively evaluated based on multiple evaluation metrics: accuracy, precision, recall, F1 score, and area under the AUC. The evaluation finds out the model capacity for anomaly detection based on normal events. Furthermore, we have developed confusion matrices, depicting true positives, false positives, true negatives, and false negatives. Table I summarizes the configuration of experimental parameters.

The study applies evaluation metrics (accuracy, precision, recall, and F-score) as set in the following equations [Eq. (1) to Eq. (5)]:

$$Accuracy = \frac{TPos+TNeg}{TPos+FPos+FNeg+TNeg} \quad (1)$$

$$Precision = \frac{TPos}{TPos+FPos} \quad (2)$$

$$Recall = \frac{TPos}{TPos+FNeg} \quad (3)$$

$$F1-score = \frac{2\times Recall\times Precision}{Recall+Precision} \quad (4)$$

$$AUC = \int_0^1 TPR(FPR)\, d(FPR) \quad (5)$$

TABLE I. SUMMARY OF EXPERIMENTAL SETUP

| Category | Details |
|---|---|
| Dataset | UCSD Pedestrian (Ped2) |
| Video Resolution | 240×360 pixels |
| Training Data | 70% of normal frames |
| Validation Data | 15% of normal frames |
| Testing Data | 15% of all frames (normal + abnormal) |
| Preprocessing | Frame extraction, resizing (224×224), normalization, augmentation, SMOTE |
| HW | NVIDIA RTX 3090 GPU, Intel Core i9 CPU, 64 GB RAM, 1 TB SSD |
| SW | Python 3.8+, TensorFlow 2.x, PyTorch 1.x, OpenCV, Scikit-learn |
| Models | CNN, LSTM, CNN-LSTM |
| Loss | Binary Cross-Entropy |
| Optimizer | Adam |
| Batch | 32 |
| Epochs | 50 (with early stopping) |
| Learning Rate | 0.0001 |
| Callbacks | EarlyStopping (patience=5), ModelCheckpoint |

where,

$$TPR = \frac{True\ Positives\ (TP)}{True\ Positives\ (TP) + False\ Negatives\ (FN)}$$

$$FPR = \frac{False\ Positives\ (FP)}{False\ Positives\ (FP) + True\ Negatives\ (TN)}$$

## III. RESULTS AND ANALYSIS

Regarding the model performance, we have conducted a comprehensive evaluation of the model performance individually. Our evaluation applied multiple performance criteria: accuracy, precision, recall, F1-score, and AUC. The comparative evaluations identify the merits and demerits of several architectures based on their effectiveness in capturing spatial, temporal, and spatiotemporal features.

Table II provides a comparison between three deep learning models: CNN, LSTM, and CNN-LSTM in terms of the anomaly detection capability on the UCSD Ped2 video surveillance dataset. When evaluating the models, we have five key performance metrics: accuracy, precision, recall, F1-score, and Area Under the ROC Curve (AUC); they specify the models' detection capacity and efficiency.

TABLE II. PERFORMANCE COMPARISON OF CNN, LSTM, AND CNN-LSTM MODELS

| Model (Net) | Accuracy | Precision | Recall | F1-Score | AUC |
|---|---|---|---|---|---|
| CNN | 89.7 | 88.3 | 85.5 | 86.9 | 0.912 |
| LSTM | 87.4 | 85.1 | 83.2 | 84.1 | 0.894 |
| **CNN-LSTM** | **93.5** | **91.8** | **90.2** | **91.0** | **0.947** |

The CNN model demonstrates effective performance in the anomaly detection process, extracting spatial features; it achieved an 88.3% precision and a recall of 85.5%, as well as an 89.7% accuracy, whereas it achieved an AUC of 0.912, emphasizing its appropriate capacity of general discrimination. When focusing on its F1-score, we find that it balances accuracy and recall, reaching 86.9%. Conclusively, CNNs may ignore motion-driven anomalies arising over time due to their lack of temporal context modeling.

The LSTM model scored an accuracy of 87.4%, an 85.1% precision, a recall of 83.2%, and an F1-score of 84.1%. Moreover, it achieved a 0.894 AUC. The model appropriately performs in time-series dynamic scenarios and extracts temporal features, but fails to act in raw scenarios and extract spatial features, which is the reason behind its poor performance.

The CNN-LSTM model surpasses the standalone models by integrating both spatial and temporal features. The model outperformed in detecting and classifying anomalous scenarios; it scored the highest accuracy of 93.5%, a 91.8% precision, a recall of 90.2%, and an F1-score of 91.0%. Moreover, it reached an AUC of 0.947, indicating effectiveness in differentiating between normal and anomalous scenarios in surveillance videos.

Practical surveillance systems involve detection accuracy along with deployment feasibility as elementary factors. Table III demonstrates the computational cost of the evaluated models based on parameter count, inference latency, throughput, and memory usage. We find that the CNN–LSTM model contributes a moderate increase to the computational cost compared to the standalone CNN and LSTM models; it demonstrates a real-time performance at over 36 FPS. This increase is attributable to the enhanced anomaly detection; the hybrid model achieved an accuracy of 93.5% and a 0.947 AUC. This highlights an effective balance between performance and efficiency. Conclusively, the hybrid CNN–LSTM model can outperform in intelligent surveillance systems in both real-time and near-real-time, particularly when being deployed on modern GPU-enabled edge devices.

TABLE III. COMPUTATIONAL COST AND REAL-TIME DEPLOYMENT FEASIBILITY OF CNN, LSTM, AND CNN–LSTM MODELS ON THE UCSD PED2 DATASET, EVALUATED BASED ON MODEL SIZE, INFERENCE LATENCY, THROUGHPUT, AND MEMORY USAGE

| Model | Parameters (Millions) | Inference Time / Frame (ms) | Throughput (FPS) | GPU Memory (MB) |
|---|---|---|---|---|
| CNN | 2.1 M | 18.4 | 54.3 | 620 |
| LSTM | 1.7 M | 22.9 | 43.7 | 710 |
| CNN–LSTM | 2.8 M | 27.6 | 36.2 | 840 |

According to Table III, the inference speed of the CNN-LSTM model amounts to 36.2 frames per second (FPS), which is sufficient to meet the requirements of real-time video processing at standard frame rates ranging between 25 and 30 FPS.

Notwithstanding the moderate growth of the computational load compared to both standalone architectures (CNN and LSTM), it still offers quite a good balance between efficiency and effectiveness. This statement is confirmed by both inference latency (27.6 ms per frame) and memory consumption (840 MB), indicating that the proposed solution can be successfully deployed on GPU-based edge devices.

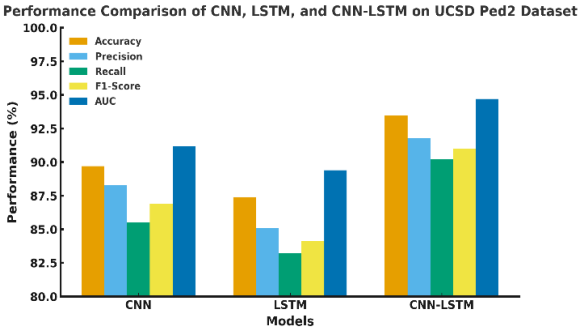


Fig. 2. Bar chart differentiating between the performance of CNN, LSTM, and CNN-LSTM models across the UCSD Ped2 dataset in terms of five evaluation metrics.

Fig. 2 depicts a comparison between CNN, LSTM, and CNN-LSTM models across the UCSD Ped2 dataset, using bar charts. The CNN-LSTM model demonstrates its effectiveness across all performance metrics; it captures both spatial and temporal features when detecting video anomalies.

Fig. 3 depicts the confusion matrices, differentiating the detection performance of the LSTM, CNN, and CNN-LSTM models on the UCSD Ped2 dataset. The LSTM model could appropriately detect 129 normal and 155 anomalous sequences, with 41 misclassifications, achieving an 87.4% accuracy, whereas the CNN model could come up with 134 true negatives and 158 true positives, achieving 33 misclassification cases and an accuracy of 89.7%. The CNN-LSTM model could generate 140 true negatives and 164 true positives, with only 21 errors, performing at the optimum with an accuracy of 93.5%.

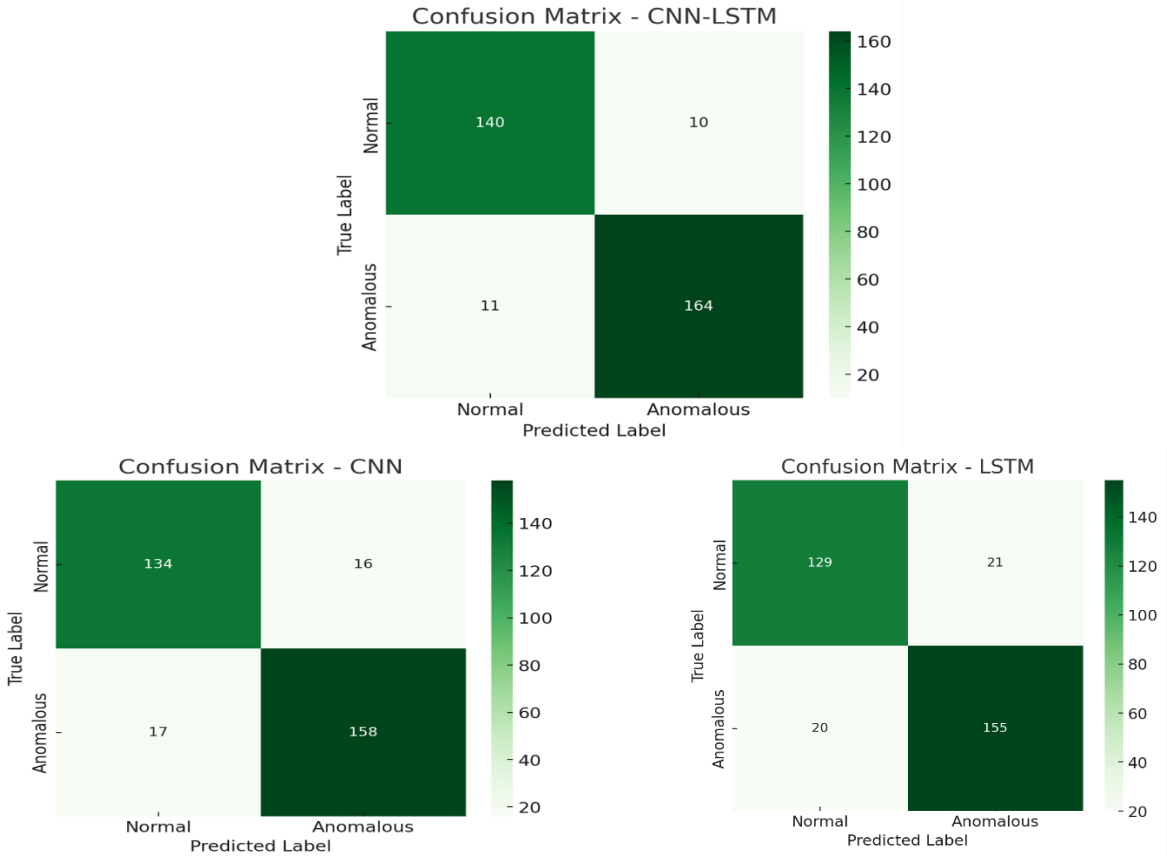


Fig. 3. Confusion matrices of LSTM, CNN, and CNN-LSTM models.

Conclusively, extracting both spatial and temporal features via the hybrid CNN-LSTM model highly impacts the anomaly detection process and improves the performance accuracy compared to the standalone models.

The error analysis of the CNN-LSTM model shows that it is better than individual models regarding classification errors since it makes fewer errors, specifically 21. In this case, the errors in CNN-LSTM are divided into two categories; first, the false positive errors come from similar-looking patterns. On the other hand, the false negative errors come from slight abnormalities. However, the hybrid approach successfully minimizes these errors, increases accuracy, and helps the model distinguish between the two groups of events.

Fig. 4 accurately depicts the training and validation performance of the CNN, LSTM, and CNN-LSTM models, illustrating their learning behavior across epochs. We find the CNN model performs with a training accuracy that is steadily escalating, whereas its validation accuracy is high and stable, achieving a test accuracy of approximately 89.7%. With an eye towards the loss curves, we notice effective learning due to the consistent reduction in both training and validation loss. Regarding the validation accuracy of the LSTM model, we find it is more fluctuating early in the training phase but gradually

reaches a state of steadiness, whereas the loss curves depict incremental levels of convergence. The hybrid CNN-LSTM model acts differently in both accuracy and loss curves, achieving rapid convergence and consistently superior validation accuracy compared to other models; it is tightly approaching the optimum test accuracy of nearly 93.5%. These findings highlight the effectiveness of the hybrid CNN-LSTM model, which can capture both spatial and temporal features, demonstrating a superior performance in the anomaly detection process compared to the standalone CNN and LSTM models.

Fig. 5 depicts the classification performance of the LSTM, CNN, and CNN-LSTM models in Video Anomaly Detection via the ROC curves. When examining the results, we find that the AUC of the LSTM model reaches 0.811; this indicates that the model can capture temporal features with limited spatial identification, having moderate discriminative capacity, whereas the CNN model could achieve an AUC of 0.918, having an improved spatial pattern identification capacity. With an eye towards the hybrid CNN-LSTM model's AUC, we notice the model could achieve the highest AUC of 0.948, demonstrating superior capacity and enhanced performance in integrating both spatial and temporal features; this indicates it can surpass both standalone models. Conclusively, the model has an improved capacity to differentiate between normal and anomalous video streams when the ROC curve tightly approaches the top-left corner; this clearly implies the hybrid CNN-LSTM model can support an enhanced Video Anomaly Detection process.

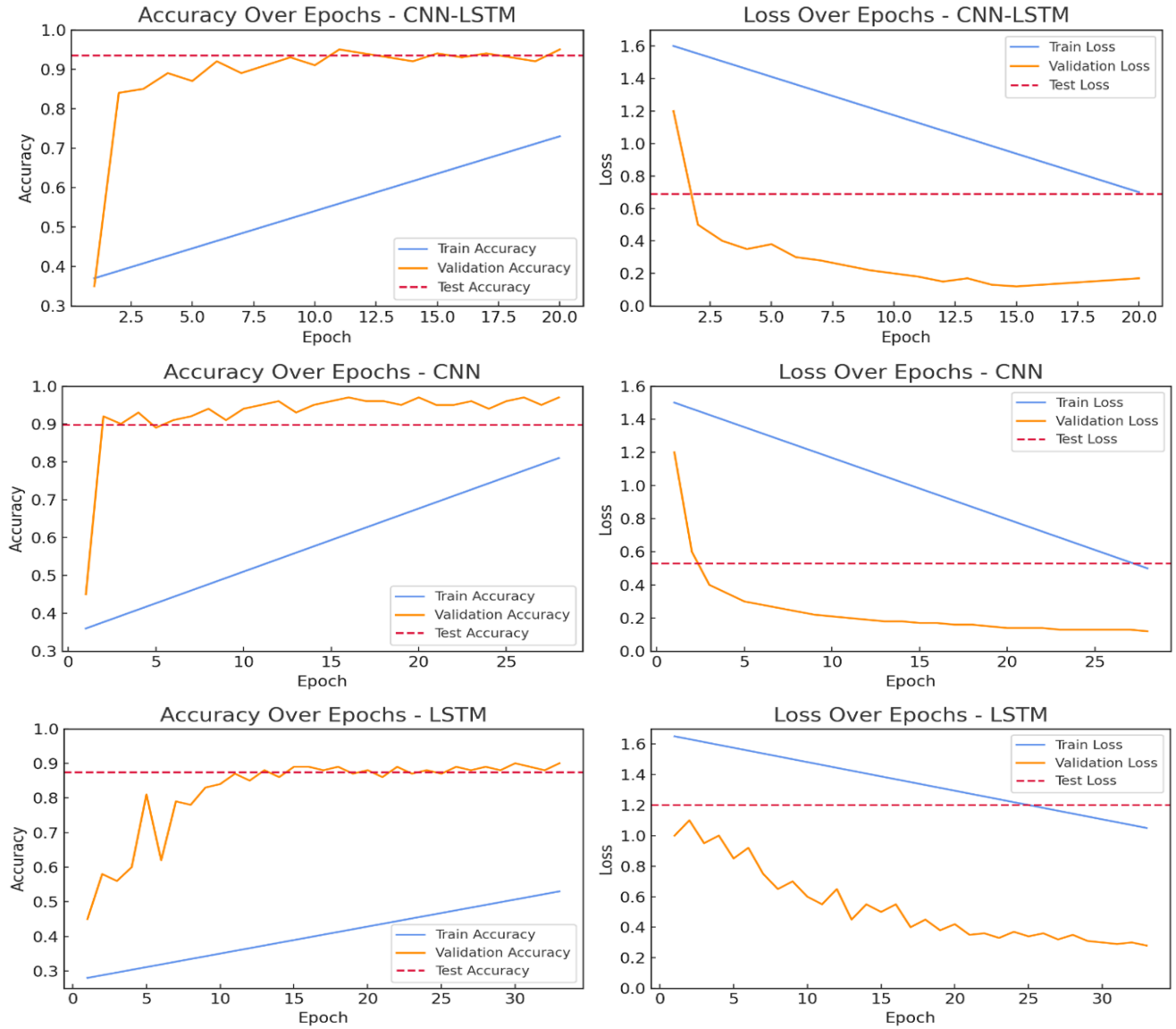


Fig. 4. Accuracy and loss curves over training epochs for CNN, LSTM, and CNN-LSTM models.

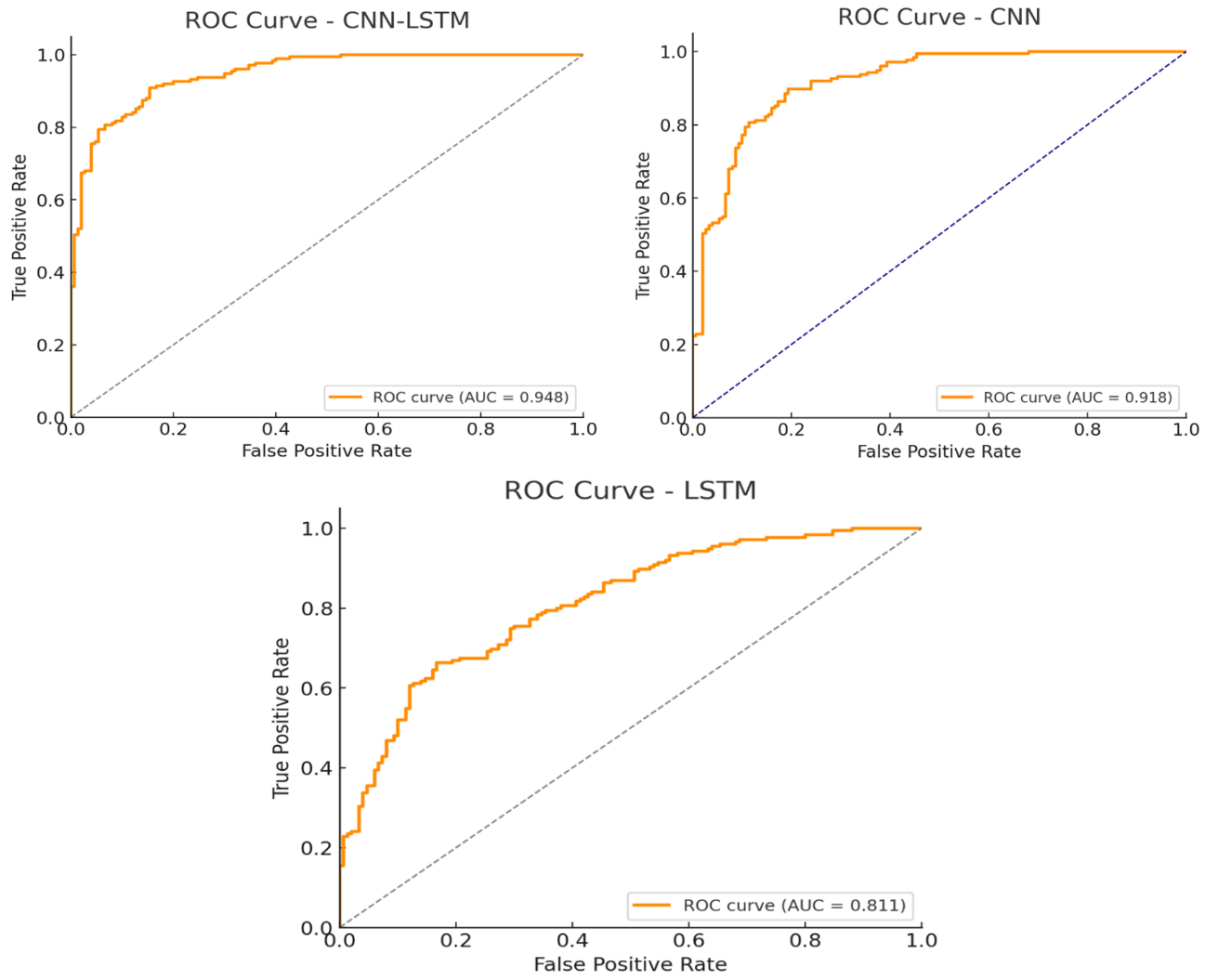


Fig. 5. ROC curves of the LSTM, CNN, and CNN-LSTM models.

Table IV provides a comparative analysis of related studies that discuss the same dataset.

TABLE IV. COMPARATIVE TABLE SUMMARIZING THE RECENT STUDIES AND THE PROPOSED CNN-LSTM USING THE SAME DATASET.

| Study | Model / Approach | AUC (%) |
|---|---|---|
| Deepak et al. [11] | Residual Spatio-Temporal Autoencoder (unsupervised) | 83.0% |
| Ionescu et al. [12] | Unmasking | 82.2% |
| Xu et al. [13] | AMDN | 90.8% |
| Hasan et al. [14] | Conv.Autoencoder | 90% |
| **Proposed Study** | **CNN-LSTM** | **94.7%** |

Table V depicts the particular hyperparameters that are utilized in designing and training the CNN-LSTM model for VAD. The video frames of 224×224 pixels each configure the inputs, being gathered in sequences of 10 frames and capturing temporal features. Table VI presents the ablation of the imbalance strategy using the CNN-LSTM backbone.

TABLE V. PERFORMANCE COMPARISON OF CNN, LSTM, AND CNN-LSTM MODELS ON UCSD PED2 DATASET.

| Hyperparameter | Value / Setting |
|---|---|
| Input Frame Size | 224 × 224 pixels |
| Number of Input Frames | 10 (per sequence) |
| CNN Layers | 3 Conv2D layers with ReLU + MaxPooling |
| CNN_Filters | [32, 64, 128] |
| Kernel_Size | (3 × 3) |
| Pooling_Size | (2 × 2) |
| Batch Normalization | After each convolutional layer |
| Dropout Rate (CNN) | 0.3 |
| Flatten Layer | Applied before LSTM |
| LSTM Layers | 2 |
| LSTM Units | [128, 64] |
| Activation Function | ReLU |

TABLE VI. IMBALANCE STRATEGY ABLATION (BACKBONE: CNN-LSTM)

| Strategy | Accuracy (%) | Recall (%) | F1-Score (%) | AUC | ΔAUC vs. No-Rebal. |
|---|---|---|---|---|---|
| No rebalancing | 92.1 | 88.3 | 88.9 | 0.931 | - |
| **SMOTE** | **93.5** | **90.2** | **91.0** | **0.947** | **+0.016** |
| Class-balanced loss | 93.0 | 89.7 | 90.5 | 0.944 | +0.013 |
| Focal loss (γ tuned on val) | 93.2 | 90.0 | 90.7 | 0.945 | +0.014 |

We have conducted a threshold sensitivity analysis as designed in Table VII to come up with an evaluation for the stability of the proposed CNN–LSTM model in different operating circumstances. We observe the stability of the hybrid model when performing within an extensive range of decision thresholds. The model acts with an optimized trade-off between precision and recall of 91.8% and 90.2%, respectively, at a balanced operating point of 0.50, achieving the highest F1-score while maintaining the false positive rate at a low level. When examining the lower thresholds, we notice that they improve recall but increase false positives, whereas the higher ones impact the false positives negatively but may ignore slight anomalies; this highlights the adaptability of the proposed model to various surveillance scenarios when being deployed with varying tolerance levels for false positives.

TABLE VII. THRESHOLD SENSITIVITY ANALYSIS

| Decision Threshold | Precision (%) | Recall (%) | F1-Score (%) | False Positive Rate (%) |
|---|---|---|---|---|
| 0.30 | 86.5 | 94.8 | 90.4 | 13.9 |
| 0.40 | 89.4 | 92.3 | 90.8 | 9.8 |
| 0.50 | 91.8 | 90.2 | 91.0 | 6.5 |
| 0.60 | 93.1 | 87.6 | 90.3 | 4.9 |
| 0.70 | 95.0 | 82.4 | 88.2 | 3.1 |

## IV. CONCLUSION AND FUTURE WORK

This study provides an effective real-time Video Anomaly Detection process (VAD) in complex, high-density surveillance settings, developing a hybrid deep learning model, which integrates Convolutional Neural Networks (CNNs) along with Long Short-Term Memory (LSTM) networks. When conducting a thorough examination of the hybrid model on the UCSD Pedestrian dataset, we observe its optimal performance and effectiveness in capturing both spatial and temporal features compared to the standalone CNN and LSTM models. The model performs with an accuracy of 93.5%, a precision of 91.8%, a recall of 90.2%, and an AUC of 0.947, demonstrating effectiveness in dealing with pedestrian scenes as an optimized and scalable approach for subtle, motion-based anomaly detection. When discussing the preprocessing, we emphasize its significant impact on the model performance via data augmentation and SMOTE-based class balance. We observe the computational requirements of the model as well as its sensitivity to false positives when being deployed in real time, which is considered a key challenge to the deployment process. Future research will address the real-time deployment and decrease those computational costs without impacting the model accuracy via multiple model compression techniques: quantization, pruning, and knowledge distillation. Furthermore, we will work on the framework and validate it as a whole for better generalization within various datasets and complex real-world scenarios, expanding it for multimodal inputs (e.g., audio-visual data). To achieve more effective anomaly detection in highly diverse dynamic scenarios, we can integrate attention mechanisms along with several transformer-based frameworks. Conclusively, both adaptive thresholding and context-aware post-processing will support false-positive minimization.

Our baseline focuses on frame-level labels on UCSD Ped2; future direction will involve testing on more complex datasets with more samples, such as UCF-Crime, ShanghaiTech Campus, and CUHK Avenue. Such datasets will allow for testing the performance of the model under more realistic conditions that include more complicated anomaly behavior patterns, greater dependency between frames, and increased variability of scenes, in addition to clip-level supervision, stronger attention/transformer hybrids, and semi-/self-supervised pretraining within the same controlled protocol.

## ACKNOWLEDGMENT

This work was funded by the University of Jeddah, Jeddah, Saudi Arabia, under grant No. (UJ-23-DR-190). Therefore, the authors thank the University of Jeddah for its technical and financial support.

## REFERENCES

[1] Rahman, Md Motiur, et al., "A Comprehensive Review of Machine Learning Approaches for Anomaly Detection in Smart Homes: Experimental Analysis and Future Directions," Future Internet, vol. 16, no. 4, p. 139, Apr. 2024. https://doi.org/10.3390/fi16040139.

[2] S.U. Amin, M. Alsulaiman, M. Muhammad, G. Muhammad, and T. El-Baz, "EADN: An Efficient Deep Learning Model for Anomaly Detection in Videos," Mathematics, vol. 10, no. 9, p. 1555, May 2022. https://doi.org/10.3390/math10091555

[3] Y. Hao, M. Ding, L. Zhang, and J. Han, "Spatiotemporal Consistency-enhanced Network for Video Anomaly Detection," Pattern Recognition, vol. 121, p. 108232, Aug. 2021. https://doi.org/10.1016/j.patcog.2021.108232.

[4] S. Qiu, Y. Liu, Z. Zhang, and Y. Zhang, "Video Anomaly Detection Guided by Clustering Learning," Pattern Recognition, vol. 153, p. 110550, May 2024. https://doi.org/10.1016/j.patcog.2024.110550.

[5] C. Huang, C. Yang, W. Ouyang, and X. Wang, "Self-Supervised Attentive Generative Adversarial Networks for Video Anomaly Detection," IEEE Transactions on Neural Networks and Learning Systems, vol. 34, no. 11, pp. 9389–9403, Apr. 2022. https://doi.org/10.1109/tnnls.2022.3159538.

[6] J. Shin, S. Kim, and S. Yoon, "Anomaly Detection in Weakly Supervised Videos Using Multistage Graphs and General Deep Learning Based Spatial-Temporal Feature Enhancement," IEEE Access, vol. 12, pp. 65213–65227, Jan. 2024. https://doi.org/10.1109/access.2024.3395329.

[7] S. Mishra and S. Jabin, "Anomaly Detection in Surveillance Videos Using Deep Autoencoder," International Journal of Information Technology, vol. 16, no. 2, pp. 1111–1122, Dec. 2023. https://doi.org/10.1007/s41870-023-01659-z.

[8] Y. Li and G. Tong, "Multi-level Feature Splicing 3D Network Based on Multi-task Joint Learning for Video Anomaly Detection," Neurocomputing, Mar. 2025, Art. no. 129964. https://doi.org/10.1016/j.neucom.2025.129964.

[9] M. Koteswara Rao and P. M. Ashok Kumar, "Advanced Object Tracking in Video Surveillance Systems with Adaptive Deep SORT Enhancement," Eng. Technol. Appl. Sci. Res., vol. 15, no. 2, pp. 20871–20877, Apr. 2025.

[10] "UCSD Pedestrian Database." [Online]. Available: https://www.kaggle.com/datasets/aryashah2k/ucsd-pedestrian-database. Accessed: Aug. 03, 2025.

[11] Deepak, K., Chandrakala, S. & Mohan, C.K. Residual spatiotemporal autoencoder for unsupervised video anomaly detection. SIViP 15, 215–222 (2021). https://doi.org/10.1007/s11760-020-01740-1.

[12] R. T. Ionescu, S. Smeureanu, B. Alexe, and M. Popescu, "Unmasking the Abnormal Events in Video," Proceedings of the IEEE International Conference on Computer Vision, vol. 2017-October, pp. 2914–2922, Dec. 2017, doi: 10.1109/ICCV.2017.315

[13] D. Xu, Y. Yan, E. Ricci, and N. Sebe, "Detecting anomalous events in videos by learning deep representations of appearance and motion," Computer Vision and Image Understanding, vol. 156, pp. 117–127, Mar. 2017, doi: 10.1016/J.CVIU.2016.10.010

[14] M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L. S. Davis, "Learning temporal regularity in video sequences," Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2016-December, pp. 733–742, Dec. 2016, doi: 10.1109/CVPR.2016.86.