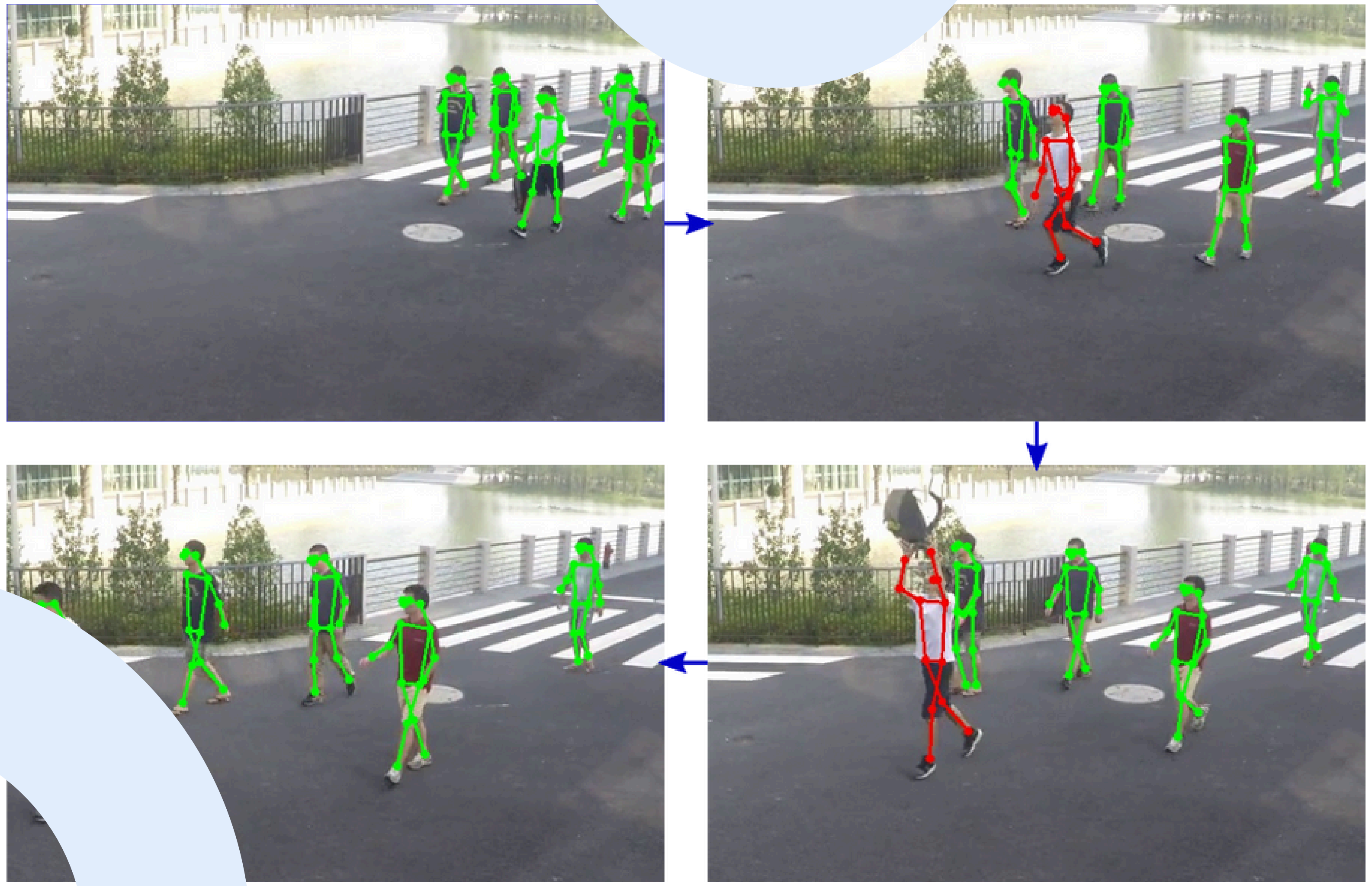
VIDEO ANOMALY DETECTION

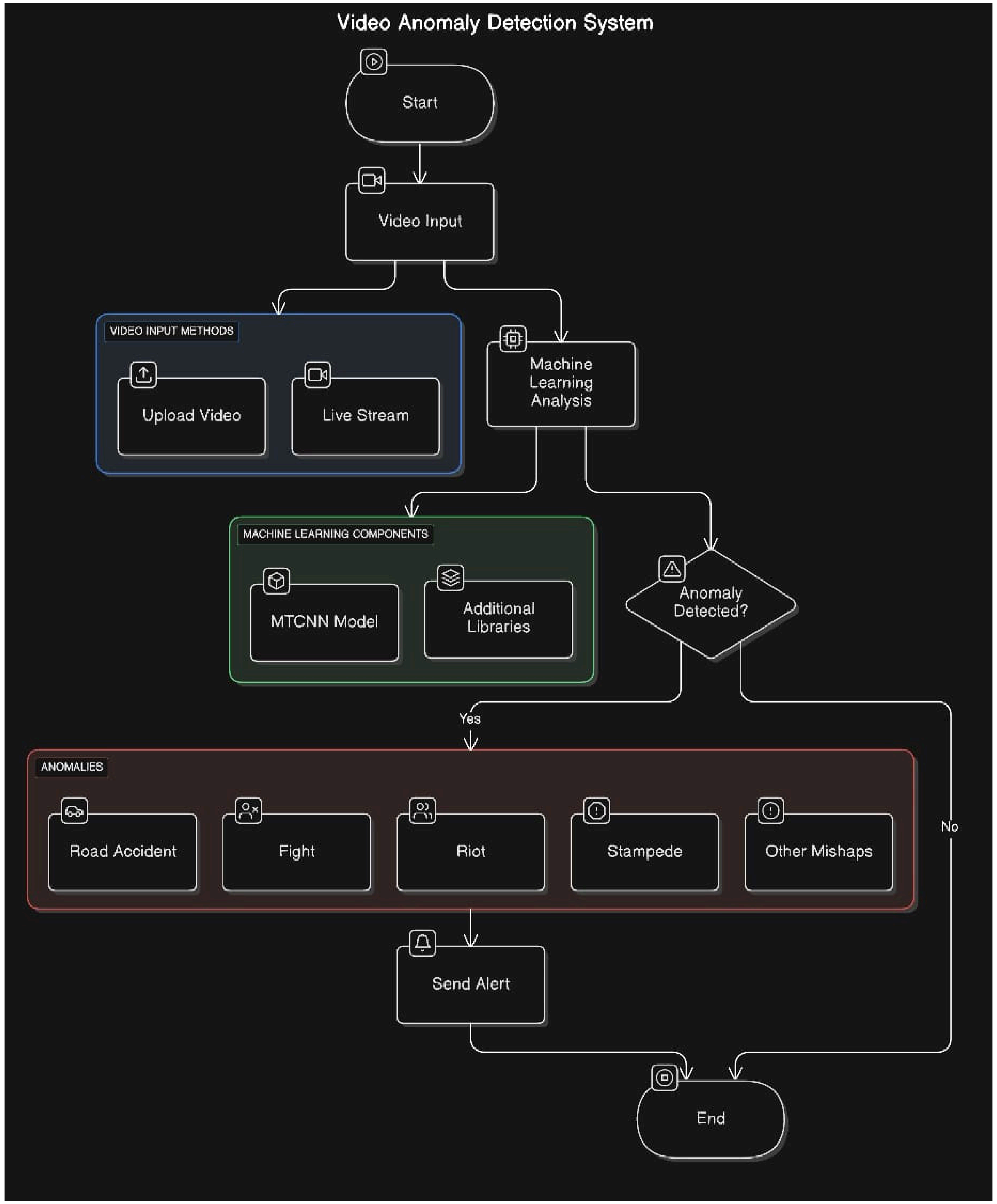
Enhancing Security and Safety through Automated Video Analysis



Introduction

Video Anomaly Detection is an advanced machine learning-based approach that enables real-time or post-event identification of unusual activities in video footage. It is particularly valuable in security, surveillance, traffic monitoring, and retail analytics, where detecting abnormal patterns can help prevent crimes, accidents, and operational inefficiencies.

With advancements in deep learning, neural networks, and real-time video processing, anomaly detection has become more accurate and scalable. The core objective of this project is to design an automated anomaly detection system that minimizes human intervention and enhances security monitoring systems.



Real-World Examples:

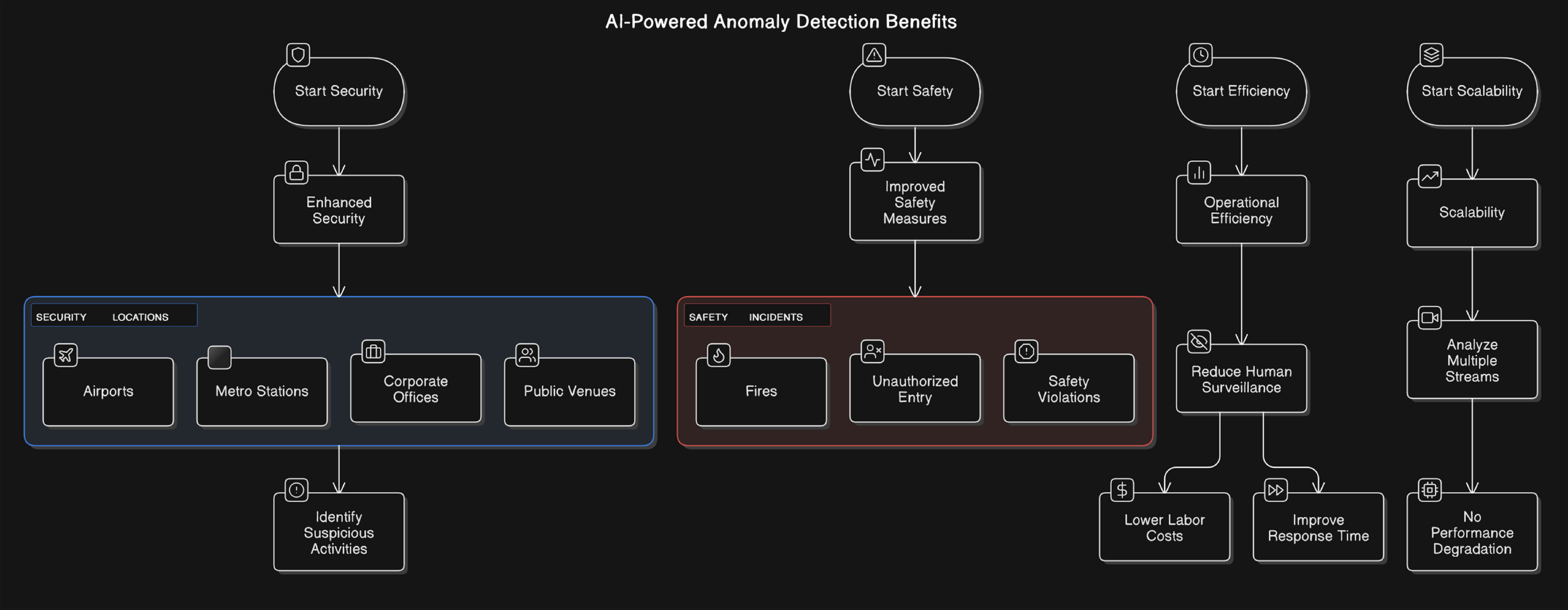
- **Security Anomalies:** Detecting unattended baggage at airports, unauthorized access in restricted areas, or suspicious movements in public spaces.
- **Traffic Anomalies:** Identifying reckless driving, sudden lane changes, congestion, and accidents in real time.
- **Retail Anomalies:** Spotting shoplifting behavior, unusual crowd gatherings, or unexpected after-hours activity in stores.
- **Industrial Safety Anomalies:** Detecting workers violating safety protocols or equipment malfunctions in hazardous environments.

Motivation for the study

The rapid increase in surveillance infrastructure worldwide has led to an overwhelming amount of video data that is challenging to monitor manually. Human operators are prone to fatigue, bias, and oversight, making it difficult to detect anomalies consistently.

A machine learning-driven anomaly detection system offers several advantages, including enhanced accuracy, real-time monitoring, and automated alerts, making security and surveillance more reliable and efficient.

- **Enhanced Security:** AI detects suspicious activities in airports, metro stations, offices, and public spaces.
- **Improved Safety:** Identifies fires, unauthorized access, and safety violations in industrial settings.
- **Scalability:** Monitors multiple video streams simultaneously without performance loss.


AI-Powered Anomaly Detection Benefits
Start Security
Enhanced Security
SECURITY LOCATIONS
Airports
Metro Stations
Corporate Offices
Public Venues
Identify Suspicious Activities
Start Safety
Improved Safety Measures
SAFETY INCIDENTS
Fires
Unauthorized Entry
Safety Violations
Start Efficiency
Operational Efficiency
Reduce Human Surveillance
Lower Labor Costs
Improve Response Time
Start Scalability
Scalability
Analyze Multiple Streams
No Performance Degradation

Problem Identification

Labor-Intensive

Requires a large workforce to monitor multiple screens continuously.

Limited Scalability

A single human operator can monitor only a limited number of cameras effectively.

Error prone

Human observers are likely to miss critical events due to fatigue, distractions, or bias.

Literature Review of Base paper

An explainable and efficient deep learning framework for video anomaly detection

Chongke Wu[1] · Sicong Shao[1] · Cihan Tunc[2] · Pratik Satam[1] · Salim Hariri[1]

Received: 20 September 2021 / Revised: 20 September 2021 / Accepted: 28 September 2021 / Published online: 23 November 2021

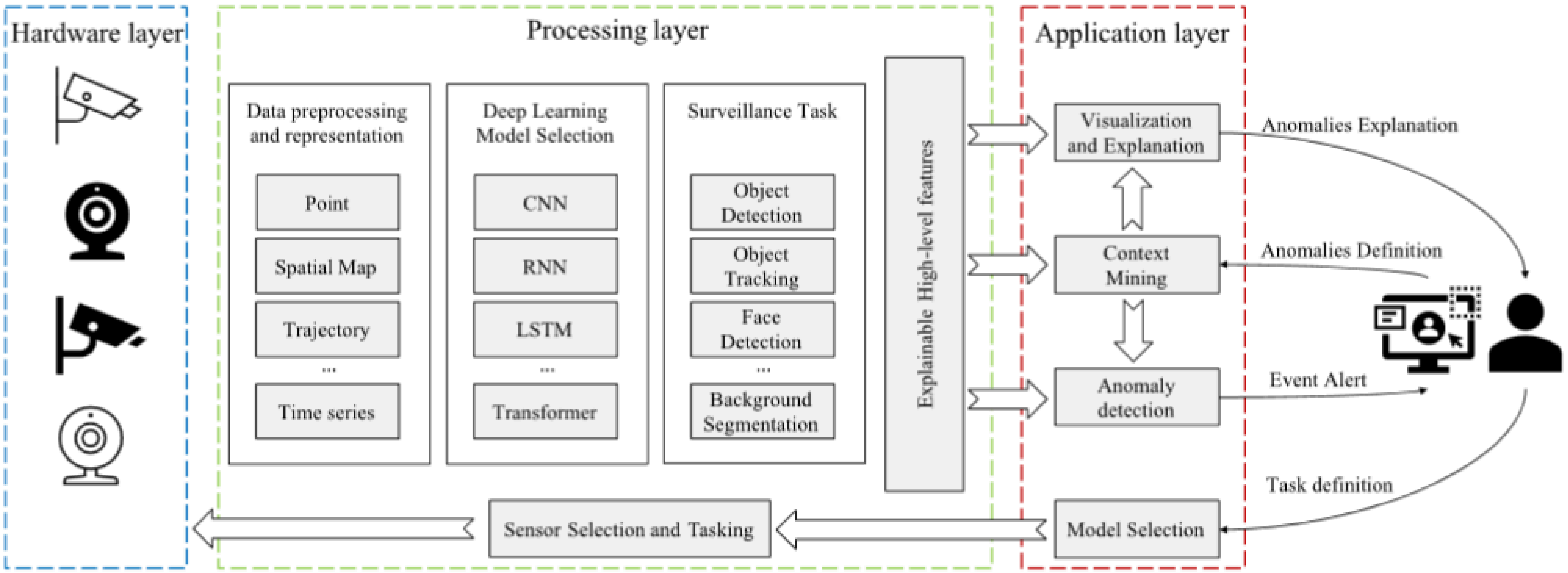
The rapid increase in surveillance infrastructure worldwide has led to an overwhelming amount of video data that is challenging to monitor manually. Human operators are prone to fatigue, bias, and oversight, making it difficult to detect anomalies consistently.

A machine learning-driven anomaly detection system offers several advantages, including enhanced accuracy, real-time monitoring, and automated alerts, making security and surveillance more reliable and efficient.

1. SUMMARY

The paper presents an explainable and efficient **deep learning framework for video anomaly detection** that overcomes challenges in interpretability and **long training times.** Existing deep learning-based methods for video anomaly detection require l**arge-scale datasets and lengthy training times,** making them difficult to deploy in real-world scenarios. Additionally, most models function as black boxes, providing little insight into their decision-making processes.

To address these issues, the proposed framework integrates **pre-trained deep models** for high-level feature extraction and utilizes a **denoising autoencoder** (**DAE**) **for anomaly detection.** The framework is capable of detecting anomalies in surveillance videos while maintaining interpretability through **SHapley Additive exPlanations** (**SHAP**)**.** The system significantly **reduces training time** (e.g., within 10 seconds on the UCSD Pedestrian datasets) while achieving performance comparable to state-of-the-art methods.



2. RESULTS

- The proposed framework achieved an AUC of 85.9% on UCSD Ped1 and 92.4% on UCSD Ped2, demonstrating its effectiveness.

- The model significantly reduced training times, requiring only 5 seconds for UCSD Ped1 and 2.9 seconds for UCSD Ped2, compared to hours for traditional models.

- The SHAP-based explainability method successfully highlighted the key features influencing anomaly detection, making the model more interpretable.

- The contextual features (such as tracking, object classification, and background segmentation) improved anomaly detection performance compared to models without them.

3. STRENGTH

- Fast Training & Deployment: The model achieves comparable accuracy to deep learning methods while training in seconds instead of hours.
- Explainability: Unlike traditional deep learning models, this framework provides insights into why an event is classified as anomalous using SHAP.
- Efficient Feature Extraction: Instead of learning low-level features from scratch, the framework leverages pre-trained CNN models for extracting high-level contextual features.
- Robust Performance: Achieves competitive AUC scores while being significantly lighter than models with complex architectures like GAN-based approaches.
- Reduced Complexity: The framework simplifies anomaly detection by separating feature extraction from decision-making, allowing for easier adaptation to different datasets.

4. LIMITATIONS

- **Dependence on Pre-trained Models:** The accuracy of the framework relies on the quality of pre-trained models, meaning poor feature extraction may negatively impact anomaly detection.
- **Limited to Frame-Level Detection:** While the model detects frame-level anomalies effectively, it does not provide pixel-level anomaly localization, which could improve precision.
- **Limited Evaluation on Other Datasets:** The method is primarily tested on UCSD Pedestrian datasets, and its effectiveness on other datasets is not extensively validated.
- **SHAP Interpretation Complexity:** Although SHAP improves explainability, interpreting the SHAP results in practical applications may still require expert knowledge.
- **Real-Time Processing Constraints:** While fast, the system still depends on pre-trained models, which could introduce latency issues in real-time surveillance.

Explanation of the problem

Video surveillance footage comprises thousands of frames per minute, making real-time monitoring extremely complex.

An effective anomaly detection system must address several challenges:

- **Defining "Normal" vs. "Abnormal" Behavior:** Establishing a baseline for normal activity is challenging since it varies across locations, times, and contexts.
- **Handling High-Volume Data:** Processing large-scale video streams requires significant computational resources and efficient deep learning models.
- **Balancing Sensitivity and Specificity:** Over-sensitive models generate excessive false alarms, while under-sensitive models may miss critical anomalies.
- **Diverse Anomaly Scenarios:** Anomalies can range from subtle behavioral deviations to sudden movements, requiring adaptable detection mechanisms.

Base paper limitation

1.Limited Dataset Generalization:

- The study only evaluates on the UCSD Pedestrian datasets (Ped1 & Ped2).
- These datasets mainly focus on pedestrian walkways, limiting the generalizability of the model to other complex scenes (e.g., crowded marketplaces, highways, subway stations).

2. Pre-trained Model Dependency:

- The framework relies on pre-trained deep models for feature extraction.
- If these pre-trained models are inaccurate, they may introduce bias and misclassification errors.

3. Lack of Real-time Processing Discussion:

- While the model achieves fast training times, the paper does not discuss real-time inference capabilities for large-scale deployments.

4. Limited Explanation for Temporal Anomalies:

- The model focuses more on spatial anomalies (e.g., detecting unusual objects in a frame).
- However, temporal anomalies (e.g., suspicious movements over time, such as loitering or erratic walking patterns) are not well-explained.

5. No Discussion on Model Robustness Against Adversarial Attacks:

- Many deep learning models are vulnerable to adversarial perturbations.
- Improvement: Investigate how the model performs against adversarial attacks and introduce defensive mechanisms (e.g., adversarial training).

Potential Improvements

1. Expand Dataset Coverage:

- Train and evaluate on multiple datasets beyond UCSD Ped1/Ped2.
- Consider more realistic, large-scale datasets with complex scenarios.

2. Fine-tune Pre-trained Models:

- Instead of relying on off-the-shelf models, fine-tune CNN-based feature extractors for better performance on target datasets.

3. Enhance Temporal Anomaly Detection:

- Introduce LSTMs, Transformer-based models, or 3D CNNs to capture anomalies in motion sequences rather than individual frames.

Methodologies to be used

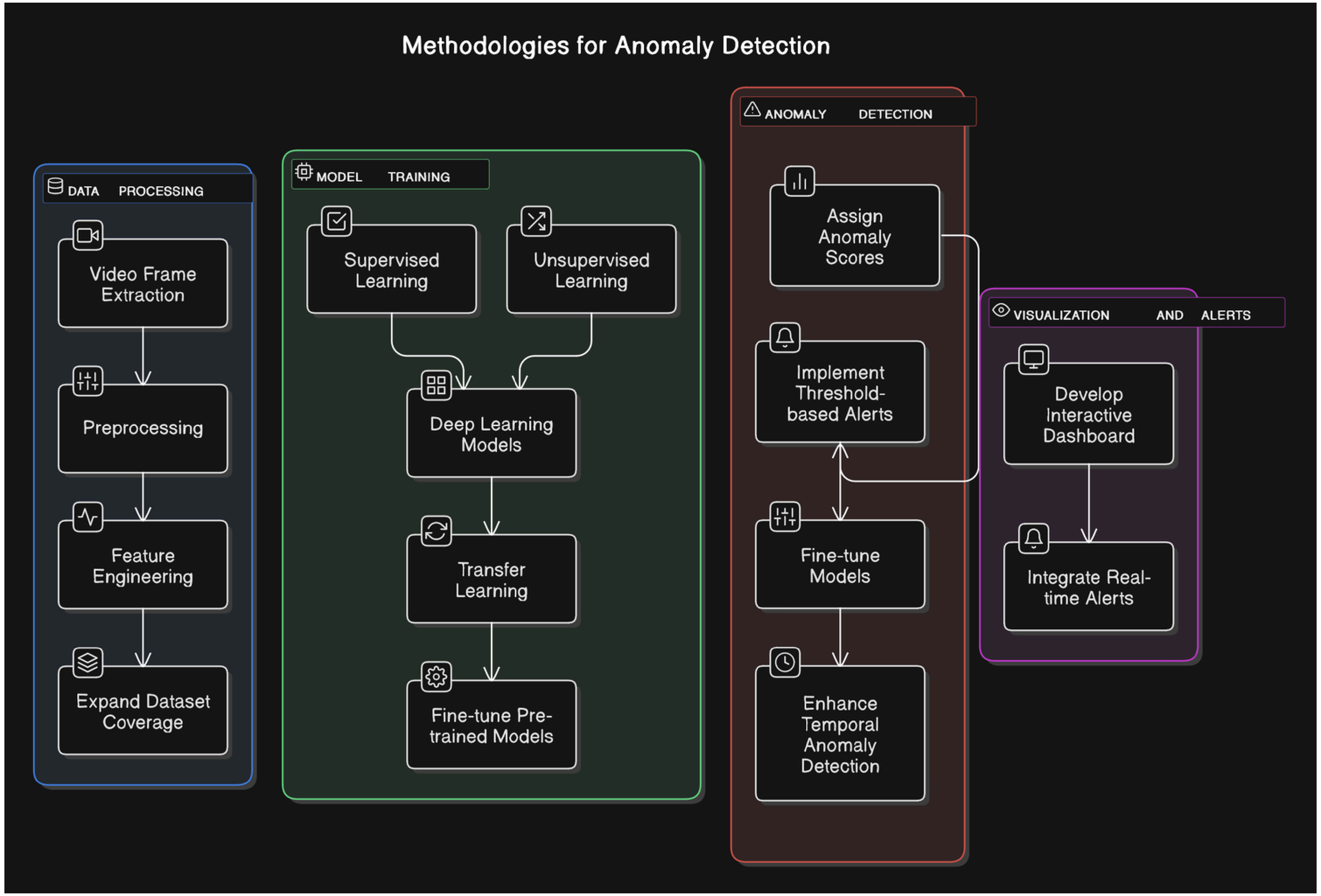
The methodology for this project follows a structured approach to developing an AI-based video anomaly detection system. It involves data preprocessing, feature extraction, model selection and training, anomaly detection, and real-time visualization.

STEP 1: Data preprocessing

STEP 2: Model training

STEP 3: Anomaly detection

STEP 4: Visualization & Alerts



1. Data Processing

- Video Frame Extraction: Using OpenCV and FFmpeg to extract frames at optimal intervals.
- Preprocessing: Image normalization, noise reduction, and frame resizing to improve model efficiency.
- Feature Engineering: Extracting movement patterns, object trajectories, and behavior sequences.
- Expand Dataset Coverage: Training and evaluating the model on multiple datasets beyond UCSD Ped1/Ped2 to improve generalizability. Considering more realistic, large-scale datasets with complex scenarios to enhance model robustness.

2. Model Training

- Supervised Learning: Training on labeled datasets where anomalies are explicitly identified.
- Unsupervised Learning: Using autoencoders and clustering algorithms to detect outliers.
- Deep Learning Models: CNNs, RNNs, and Transformer-based architectures for learning spatial-temporal patterns.
- Transfer Learning: Using pre-trained models like YOLO or MobileNet for faster and more accurate anomaly detection.
- Fine-tune Pre-trained Models: Instead of relying on off-the-shelf models, fine-tuning CNN-based feature extractors to achieve better performance on target datasets.

3. Anomaly Detection

- Assigning anomaly scores based on deviations from learned patterns.
- Implementing threshold-based alerting mechanisms.
- Fine-tuning models to minimize false positives and false negatives.
- Enhance Temporal Anomaly Detection: Introducing LSTMs, Transformer-based models, or 3D CNNs to capture anomalies in motion sequences rather than individual frames.

4. Visualization & Alerts

- Developing an interactive dashboard for live video monitoring.
- Integrating real-time alerts and notifications for detected anomalies.

Base Paper Result

The base paper reports the following AUC (Area Under the Curve) scores for video anomaly detection:

UCSD Ped1 Dataset: 85.9%

UCSD Ped2 Dataset: 92.4%

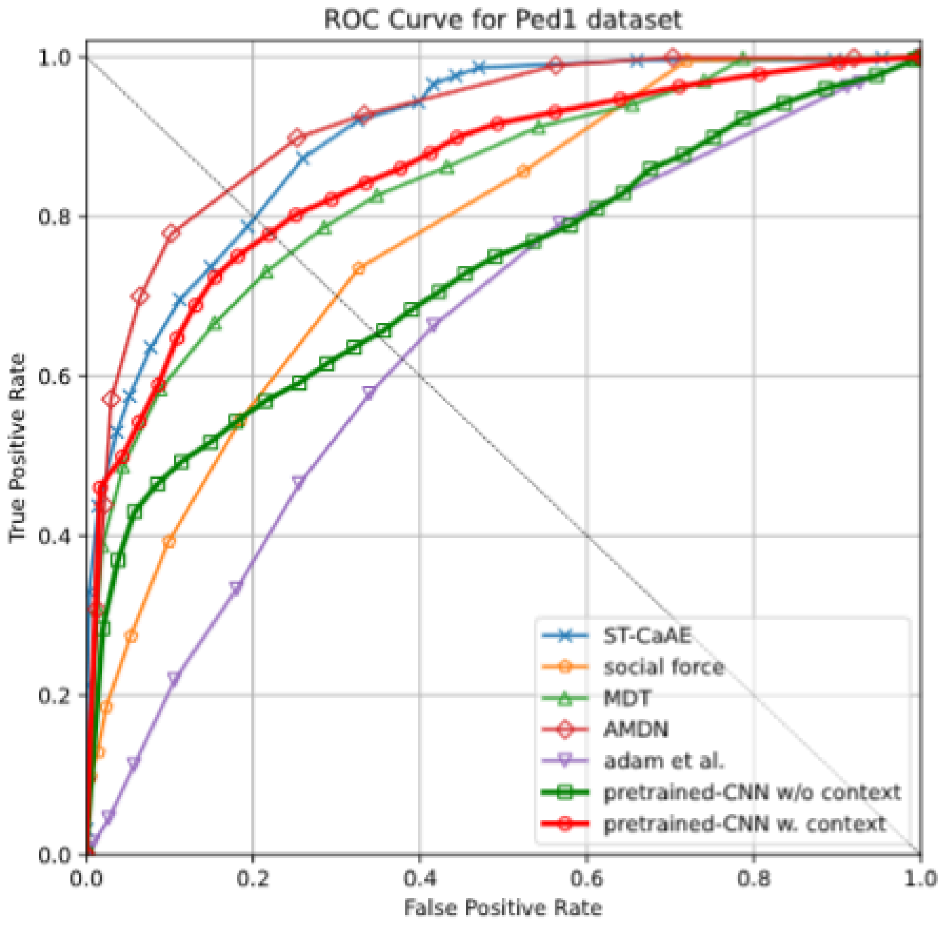


Fig. 7 ROC curve of Ped1 dataset

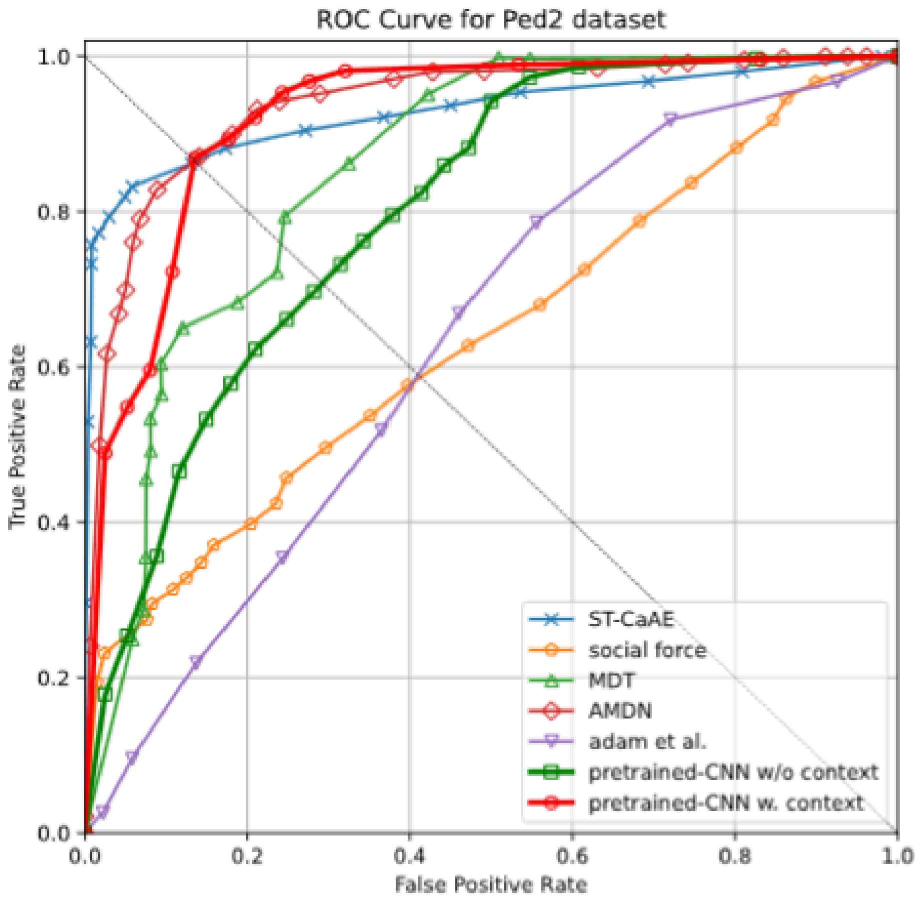


Fig. 8 ROC curve of Ped2 dataset

Expected Improvements with Your Enhancements

With all these enhancements, a realistic goal would be:

- UCSD Ped1: 90-93% AUC (compared to 85.9%)
- UCSD Ped2: 94-96% AUC (compared to 92.4%)
- On larger datasets like Avenue or ShanghaiTech: Expect 85-90% AUC, which would be competitive with state-of-the-art models.

Optimizing Real-Time Performance:

- The base paper does not discuss real-time inference speed, which is crucial for deployment.
- By optimizing inference, the model could be 2-5x faster, enabling real-time anomaly detection.

Expanding to more diverse datasets

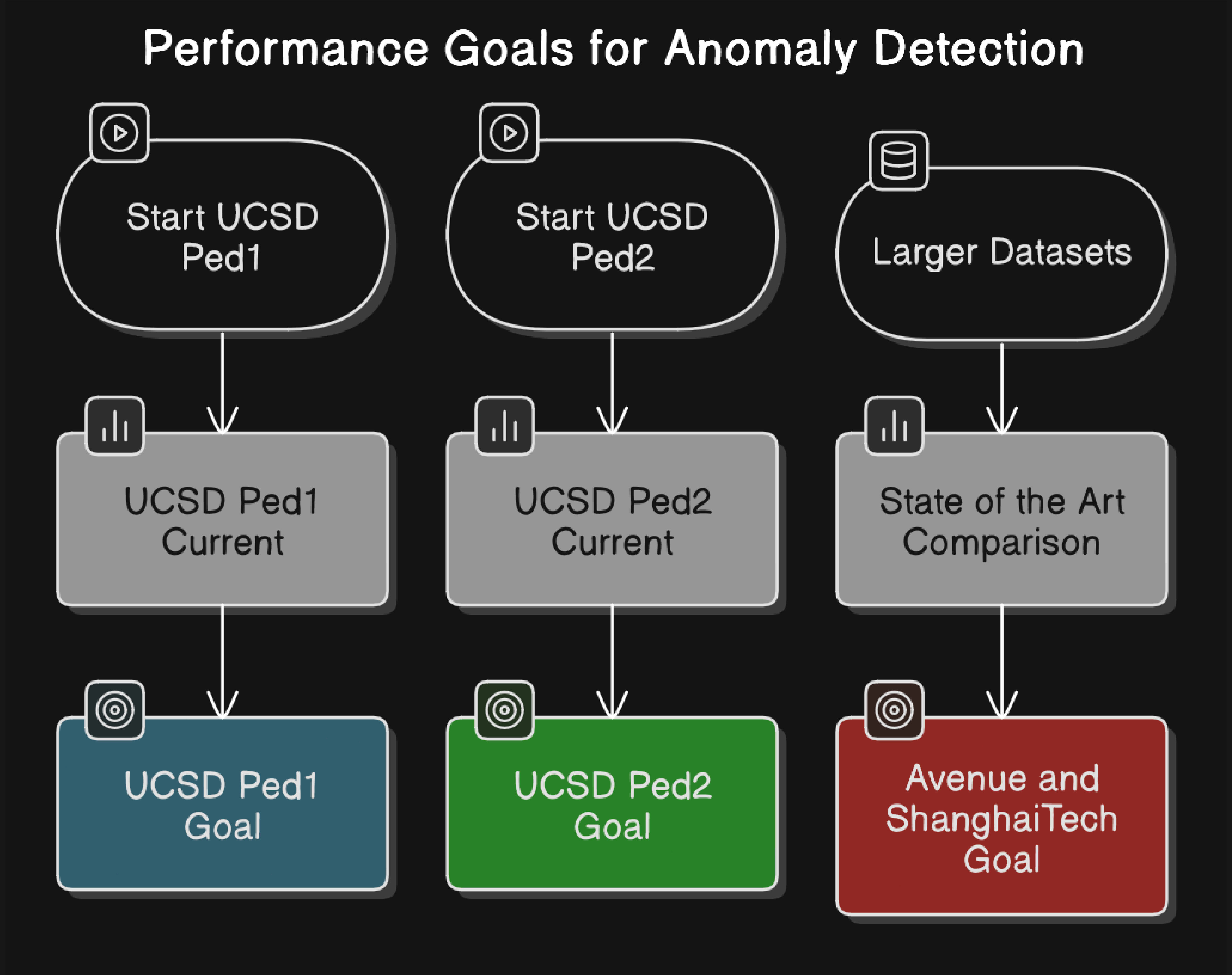
- Base Paper was tested only on UCSD Ped1/Ped2, which have limited real-world complexity.
- Using datasets like ShanghaiTech, Avenue, or UCF-Crime could improve generalization and increase AUC by 2-5%.

Fine-tuning Pre-trained Models

- The base model relies on pre-trained CNNs without adapting them to the dataset.
- Fine-tuning could reduce false positives and improve feature extraction accuracy, leading to a 1-3% AUC boost.

Timeline of the events

Task	Duration
Data Collection & Preprocessing	2 weeks
Model Selection & Training	4 weeks
System Integration & Testing	3 weeks
Evaluation & Refinement	2 weeks
Deployment & Final Testing	2 weeks


Performance Goals for Anomaly Detection
Start UCSD Ped1
Start UCSD Ped2
Larger Datasets
UCSD Ped1 Current
UCSD Ped2 Current
State of the Art Comparison
UCSD Ped1 Goal
UCSD Ped2 Goal
Avenue and ShanghaiTech Goal

Workflow and Setup of the Project

`src/organize_and_process.py` → Initial features and metadata.

EDA Notebook (optional) → Data exploration and validation.

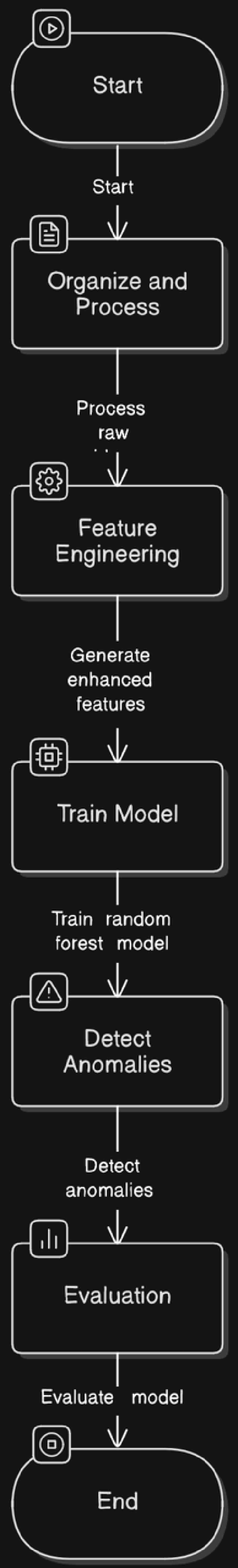
`src/feature_engineering.py` → Enhanced feature extraction.

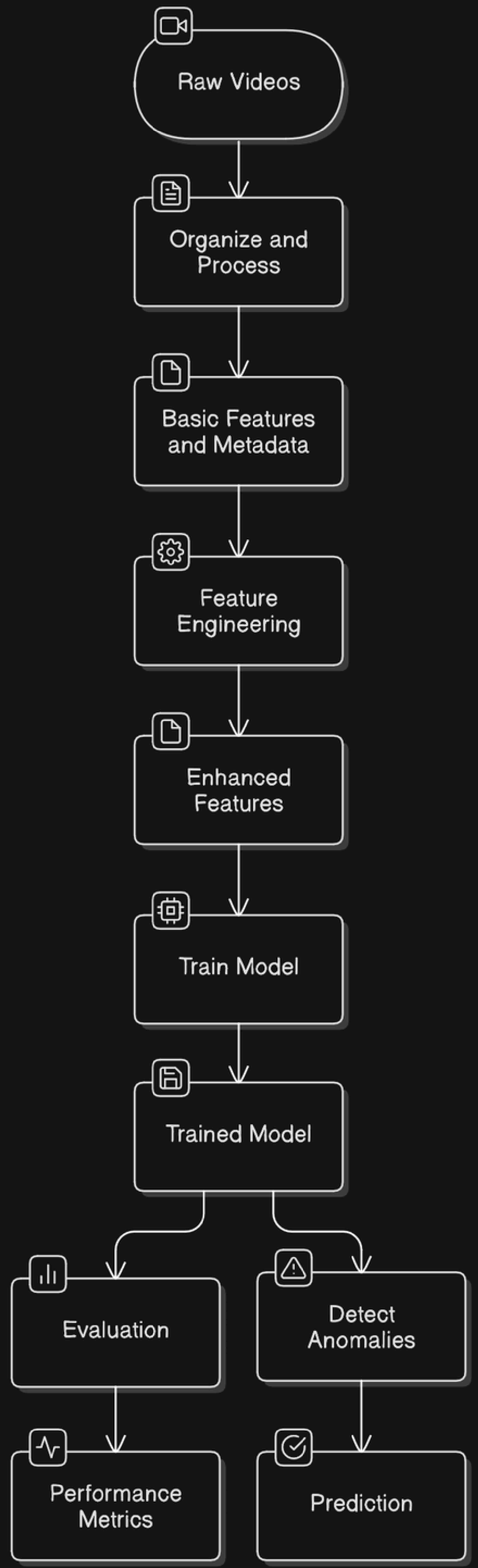
`src/train_model.py` → Model training.

`src/detect_anomalies.py` → Anomaly prediction on new videos.

`src/evaluation.ipynb` → Performance evaluation.

```
video-anomaly-detection/
├── data/
│   ├── raw_videos/                # Raw video files
│   │   ├── anomaly/               # Anomaly videos (e.g.,
Abuse027_x264.mp4)
│   │   └── normal/                # Normal videos (e.g.,
Normal_Videos_010_x264.mp4)
│   ├── processed/                 # Processed data
│   │   ├── features/              # .npy files with basic motion features
│   │   ├── frames/                # Extracted frames (if used)
│   │   └── metadata/              # Metadata files
│   │       ├── video_metadata.csv  # Video info from initial processing
│   │       └── enhanced_features.csv  # Final features for training
├── Sample/                        # Sample videos shot from phone for
testing
│   ├── sample_anomaly_1.mp4       # Example anomaly video
│   ├── sample_normal_1.mp4        # Example normal video
│   └── ...                        # Add more as needed
├── src/                           # Source code
│   ├── organize_and_process.py    # Step 1: Initial feature extraction
│   ├── feature_engineering.py     # Step 2: Enhance features with optical
flow
│   ├── train_model.py             # Step 3: Train the model
│   ├── detect_anomalies.py        # Step 4: Predict on new videos
│   └── evaluation.ipynb           # Step 5: Visualize results
├── models/                        # Trained model
│   └── rf_model.pkl               # Random Forest model
└── README.md                      # This file
```





Evaluation Metrics and Visualizations

Evaluation Metrics used

The following evaluation metrics have been used to determine the usefulness and accuracy of the ML-model trained:

1. Precision
2. Recall
3. F1 Score
4. Support
5. Macro Average
6. Weighted Average
7. Accuracy

Macro Average and Weighted Average

- Macro averages of 0.80 (precision), 0.81 (recall), and 0.80 (F1-score) provide a balanced view, emphasizing the model's performance across both classes without favoring the majority
- Weighted averages of 0.80 (precision), 0.81 (recall), and 0.82 (F1-score) reflect the model's performance weighted by the larger anomalous class (45 samples).

Precision, Recall, F1-Score and Support

- For class 0 (normal), a precision of 0.71 means that 71% of the segments predicted correctly normal. For class 1 (anomalous), a precision of 0.88 means that 88% of the segments predicted as anomalous were correct.
- For class 0, a recall of 0.80 was recorded for normal and for class 1, 0.82 was recorded as the recall for the anomalies.
- For class 0, an F1-score of 0.75 reflects a balanced performance between precision and recall. For class 1, an F1-score of 0.85 indicates a reasonable balance.
- With 25 normal and 45 anomalous samples, the dataset is moderately imbalanced, with anomalies being more frequent.

Accuracy: 0.81 (81%)

Classification Report of the trained model

```
... Classification Report:
              precision    recall  f1-score   support

           0       0.71      0.80      0.75        25
           1       0.88      0.82      0.85        45

    accuracy                           0.81        70
   macro avg       0.80      0.81      0.80        70
weighted avg       0.82      0.81      0.82        70


Confusion Matrix:
[[20  5]
 [ 8 37]]
```

With 20 TN, 5 FP, 8 FN, and 37 TP, the model excels at detecting anomalies (37/45 correct) but misses 8 anomalies and misclassifies 5 normal segments, indicating a need to refine sensitivity or specificity

Metrics(from image):

- Class 0 (Normal): Precision = 0.71, Recall = 0.80, F1-score = 0.75, Support = 25.
- Class 1 (Anomalous): Precision = 0.88, Recall = 0.82, F1-score = 0.85, Support = 45.
- Accuracy: 0.81.
- Macro Avg: Precision = 0.76, Recall = 0.84, F1-score = 0.75.
- Weighted Avg: Precision = 0.80, Recall = 0.81, F1-score = 0.82.

Calculations:

- Precision = TP / (TP + FP).
- Recall = TP / (TP + FN).
- F1-score = 2 * (Precision * Recall) / (Precision + Recall).
- Accuracy = (TP + TN) / Total.
- Macro Avg: Unweighted mean across classes.
- Weighted Avg: Support-weighted mean.

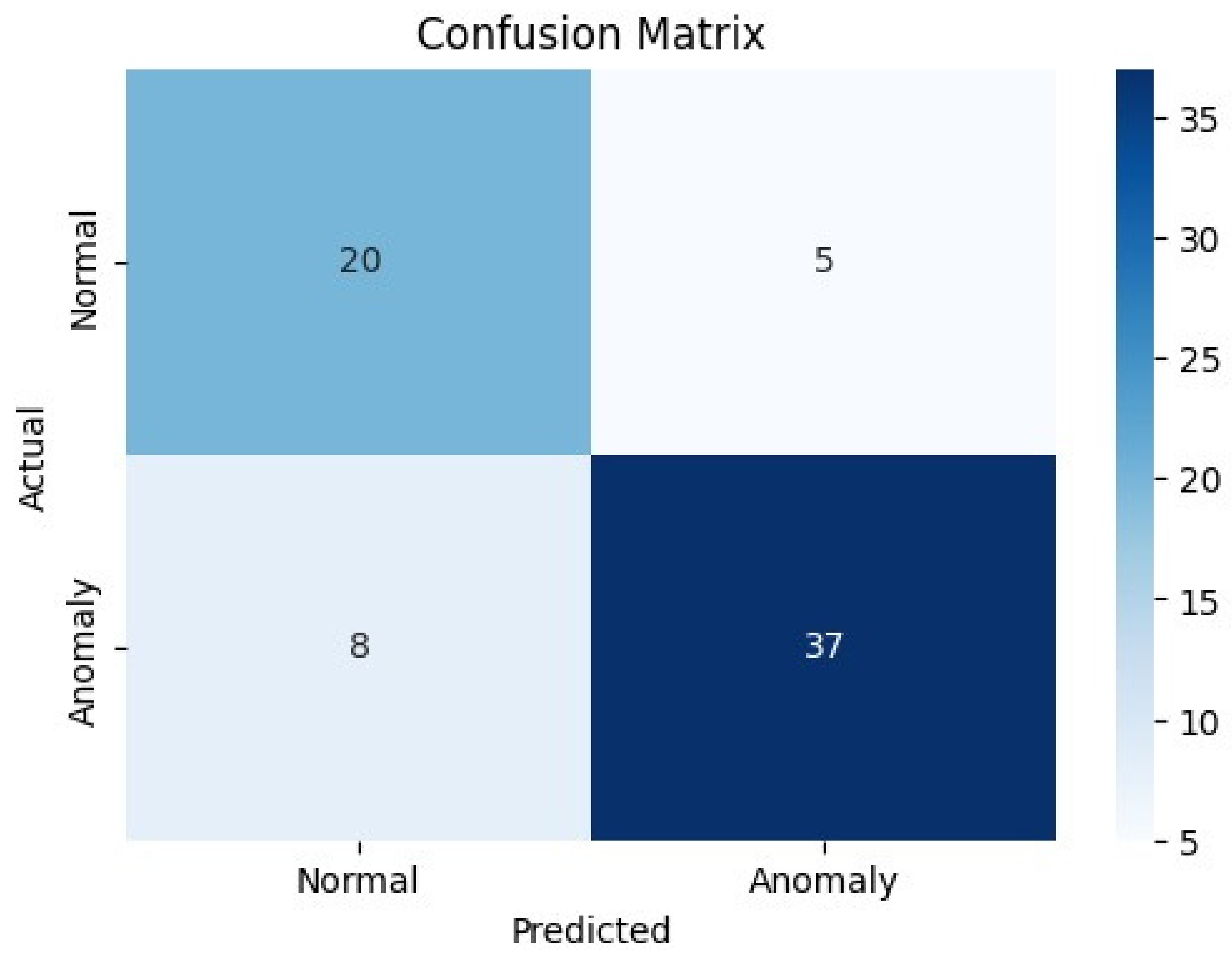
Confusion Matrix

It contains four key values:

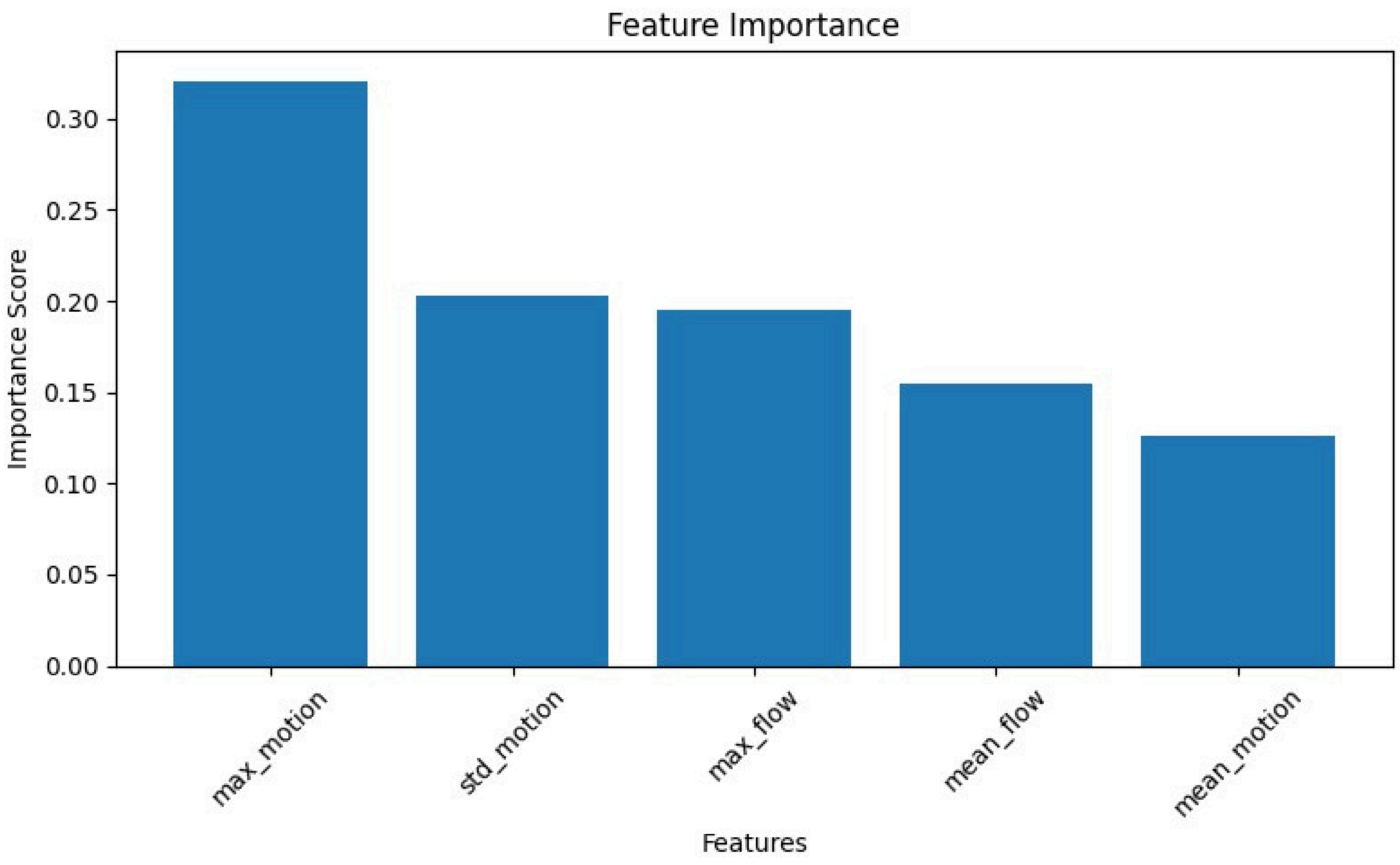
- **True Negatives (TN): Correctly predicted normal instances.**
- **False Positives (FP): Normal instances incorrectly predicted as anomalous.**
- **False Negatives (FN): Anomalous instances incorrectly predicted as normal.**
- **True Positives (TP): Correctly predicted anomalous instances.**

The model performs exceptionally well for the anomalous class (class 1), with 37 true positives out of a possible total (37 + 8 = 45), indicating a high detection rate. However, it misclassifies 8 anomalous segments as normal, suggesting some missed anomalies. For the normal class (class 0), it correctly identifies 20 out of 25 instances (20 + 5), but the 5 false positives indicate some overprediction of anomalies. This suggests a strong overall performance

- **True Negatives (TN): 20 - Correctly predicted normal segments.**
- **False Positives (FP): 5 - Normal segments incorrectly predicted as anomalous.**
- **False Negatives (FN): 8 - Anomalous segments incorrectly predicted as normal.**
- **True Positives (TP): 37 - Correctly predicted anomalous segments.**



Feature Importance Score Graph



The model prioritizes motion-related features, especially the average motion (mean_motion), over flow-related features to detect anomalies. This suggests that unusual motion patterns—such as sudden bursts or prolonged stillness—are strong indicators of anomalies in video data, which aligns with expectations for tasks like detecting arson or irregular activities.

- Each bar represents a feature (e.g., mean_motion, max_motion, std_motion, mean_flow, max_flow), and its height reflects the importance score, usually on a scale from 0 to 1. A higher score means the feature has a greater influence on the model's predictions.

Purpose:

- It reveals which features are most critical for distinguishing between normal and anomalous video segments. This helps validate whether the model's reliance on certain features aligns with domain knowledge

End Result:

- A typical result might show mean_motion with the highest importance score (e.g., ~0.35), followed by max_motion (~0.25), std_motion (~0.15), mean_flow (~0.15), and max_flow (~0.10).

Precision-Recall Curve

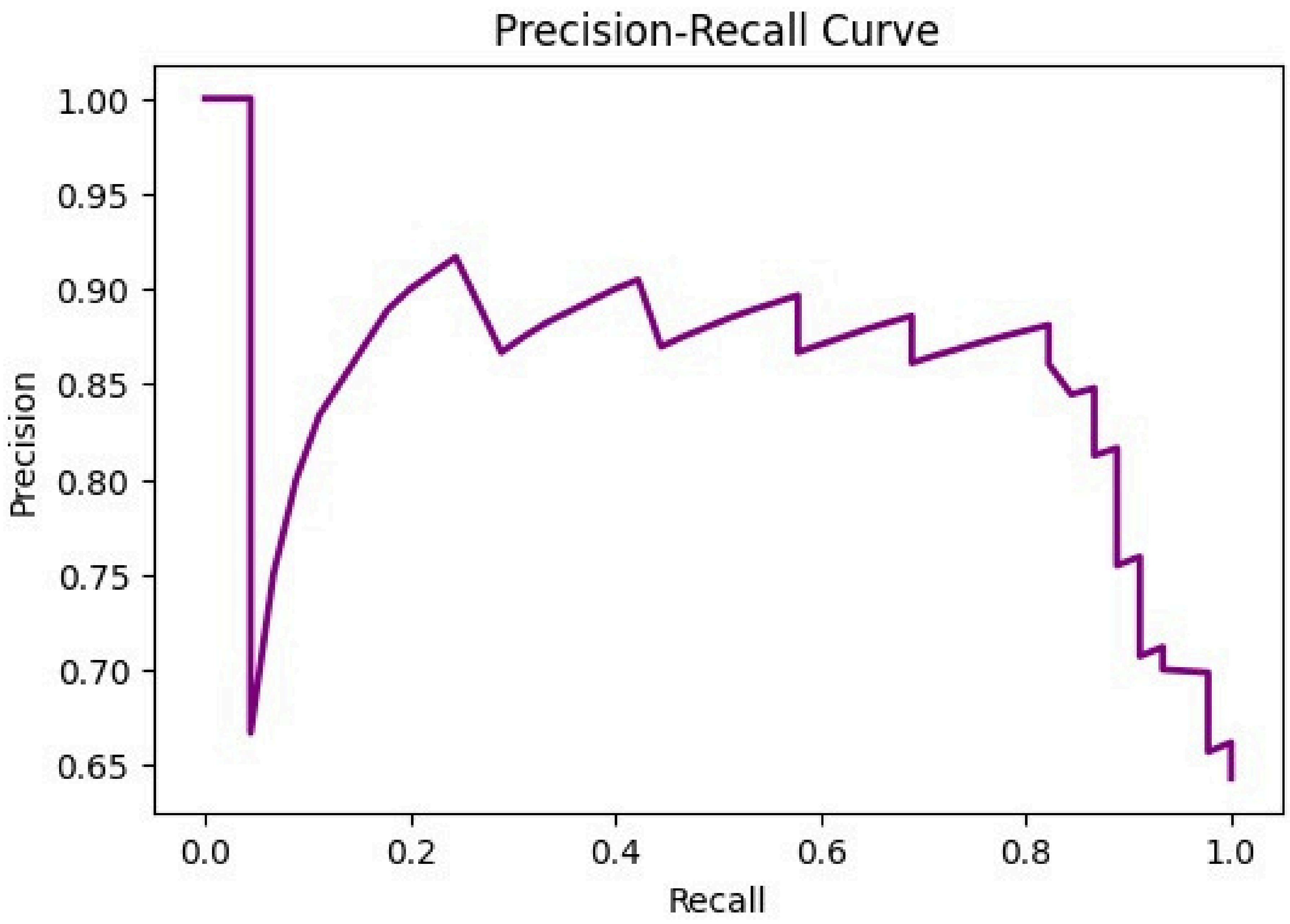
Significance:

- Shows the trade-off between precision and recall, assessing how well the model balances accurate anomaly predictions with comprehensive detection.

End Result:

- Precision starts high (~0.9) at low recall (~0.2) and declines (e.g., to ~0.3) as recall approaches 1, indicating strong performance with conservative thresholds but difficulty detecting all anomalies without false positives.
- The curve illustrates the trade-off between these two metrics: increasing recall (detecting more anomalies) often reduces precision (due to more false positives).

It evaluates the model’s performance across various decision thresholds, which is especially useful for imbalanced datasets where anomalies are rare compared to normal segments. The area under the curve (AUC) can summarize overall performance, with a higher AUC indicating a better balance between precision and recall.



ROC Curve

An AUC of 0.82 indicates good discriminative ability, meaning the model is 82% better than random guessing at distinguishing between normal and anomalous video segments. The steep initial rise suggests high sensitivity at low false positive rates, but the flattening indicates that further increases in TPR come with a higher FPR, reflecting a trade-off as the threshold adjusts. This aligns with the strong performance on the anomalous class (37 TP) and moderate false positive issues (5 FP) seen in the confusion matrix.

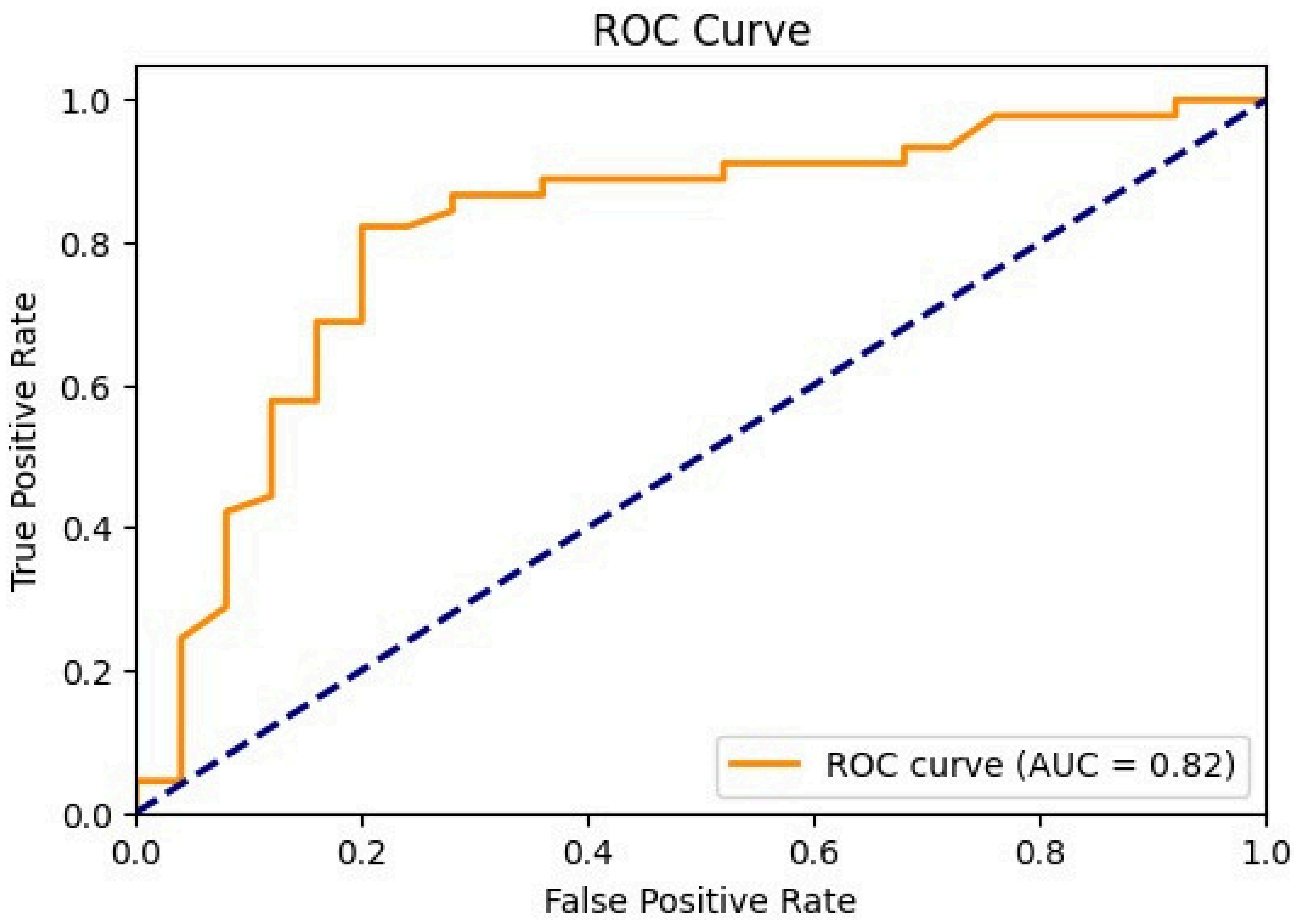
- The Area Under the Curve (AUC) quantifies the overall ability of the model to distinguish between classes, with a value closer to 1 indicating excellent performance and 0.5 indicating no discriminative power.

Purpose:

- The ROC curve assesses the model's ability to balance sensitivity (detecting anomalies) and specificity (correctly identifying normal segments) across different thresholds. It is particularly valuable for imbalanced datasets, as it provides a threshold-independent measure of performance.

Calculations:

- TPR = TP / (TP + FN) – The proportion of actual positives correctly identified.
- FPR = FP / (FP + TN) – The proportion of actual negatives incorrectly identified as positive.



Exploratory Data Analysis of the Dataset

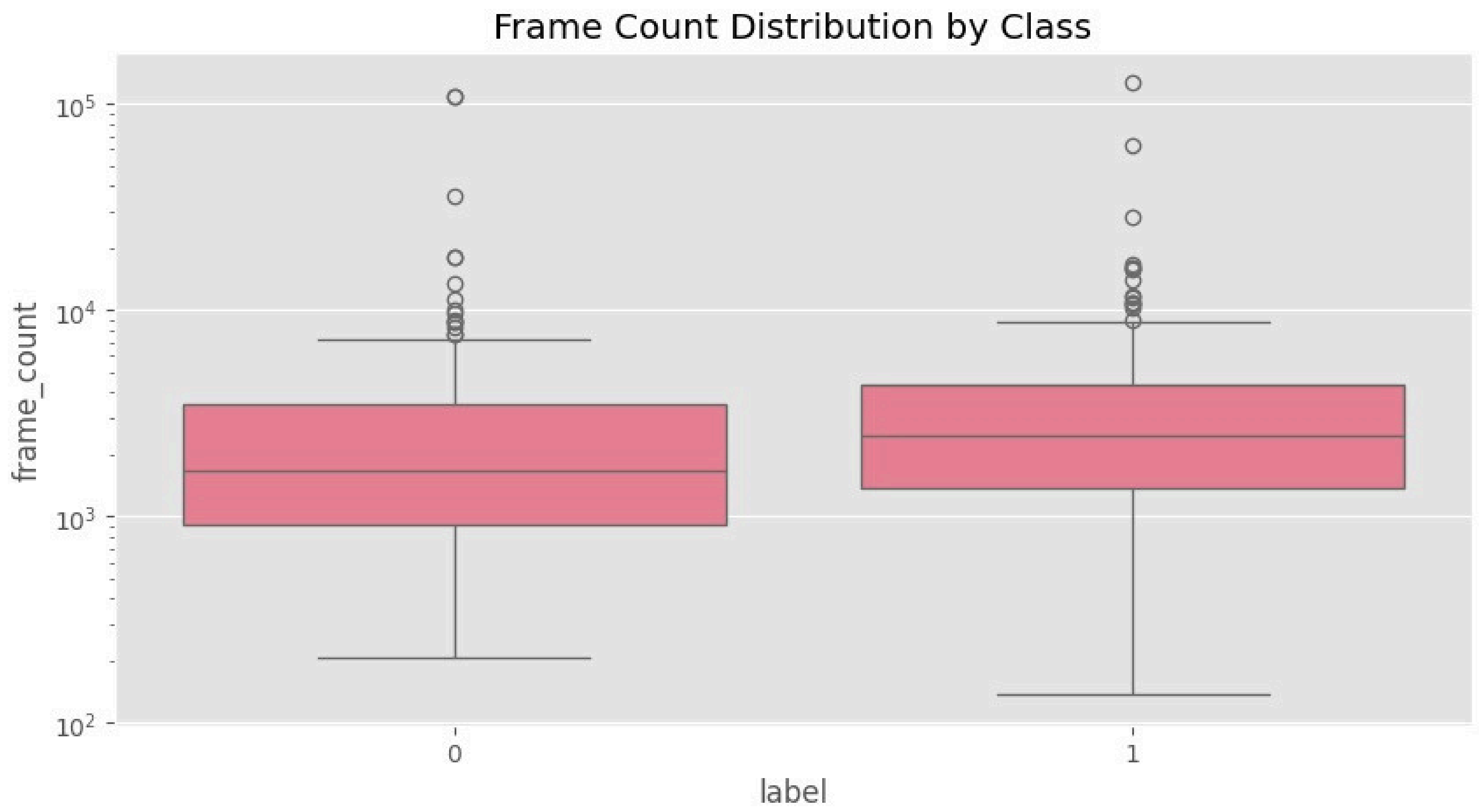
Frame Count Distribution by Class

Insights Conveyed:

- The boxplot conveys that video length alone may not be a definitive indicator of crime but highlights variability. For instance, a lower median frame count for anomalous videos (e.g., ~1,133 frames) compared to normal (~2,028 frames) might suggest that many crimes are brief or captured in shorter clips, such as a sudden Robbery. Conversely, outliers with high frame counts (e.g., 76,988 frames in an Arson case) indicate that some crimes, like prolonged fires, extend over time, reflecting the dataset's diversity.

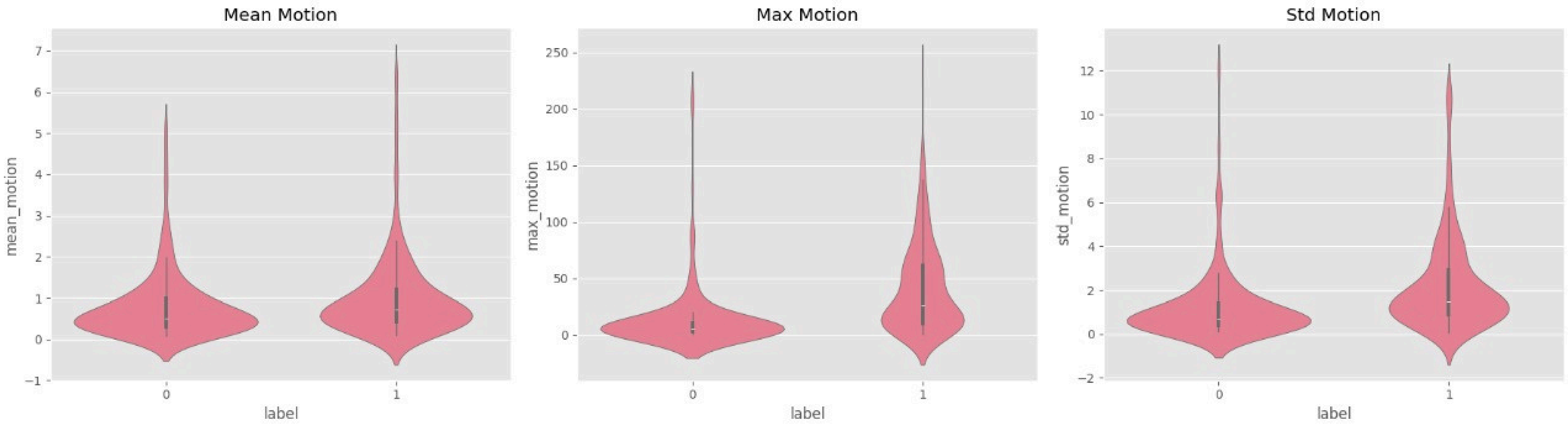
Purpose

- To assess whether video length, as indicated by frame count, varies systematically between normal surveillance footage and crime-related anomalous events, potentially aiding in identifying temporal patterns of criminal activity.



This boxplot visualizes the distribution of frame counts for the two classes in the UCF Crime dataset: normal (label = 0) and anomalous (label = 1, encompassing crime types like Arson or Robbery). The x-axis represents the class, and the y-axis displays frame counts on a logarithmic scale to manage the dataset's wide range of video lengths (e.g., from a few hundred to over 76,000 frames). The plot includes medians, interquartile ranges (IQR), and outliers, offering a statistical overview of video duration.

Motion Features Comparison (Violin Plots)



This figure features three violin plots comparing the distributions of motion features—mean motion (average intensity), maximum motion (peak intensity), and standard deviation of motion (variability)—across normal and anomalous videos in the UCF Crime dataset. The x-axis denotes the class, and the y-axis shows feature values, with violin plots blending boxplot statistics (median, IQR) with density curves to reveal data spread.

Motion Features Comparison (Violin Plots)

Purpose

- To investigate how motion characteristics differ between normal surveillance footage and crime events, leveraging motion as a primary cue for detecting anomalies like fights or explosions in the UCF Crime dataset.

Insights Conveyed:

- Mean Motion: The plot conveys that average motion intensity is slightly higher for anomalous videos, suggesting baseline activity (e.g., crowd movement during a Riot) differs from the steady pedestrian flow in normal footage. However, overlapping distributions imply mean motion alone is insufficient for crime detection.
- Max Motion: A broader, right-skewed distribution for anomalies delivers a strong message: peak motion events (e.g., a sudden Assault or Explosion) are more pronounced in crimes, with spikes reflecting abrupt, violent actions. This highlights max motion as a key discriminator.
- Std Motion: Greater variability in anomalies indicates irregular motion patterns (e.g., erratic movements in a Burglary), conveying that crime often involves unpredictable behavior, unlike the consistent motion in normal scenes.
- Motion features, especially max and std motion, are critical for identifying crime, but their overlap with normal data suggests combining them with contextual features (e.g., object detection) for robustness.

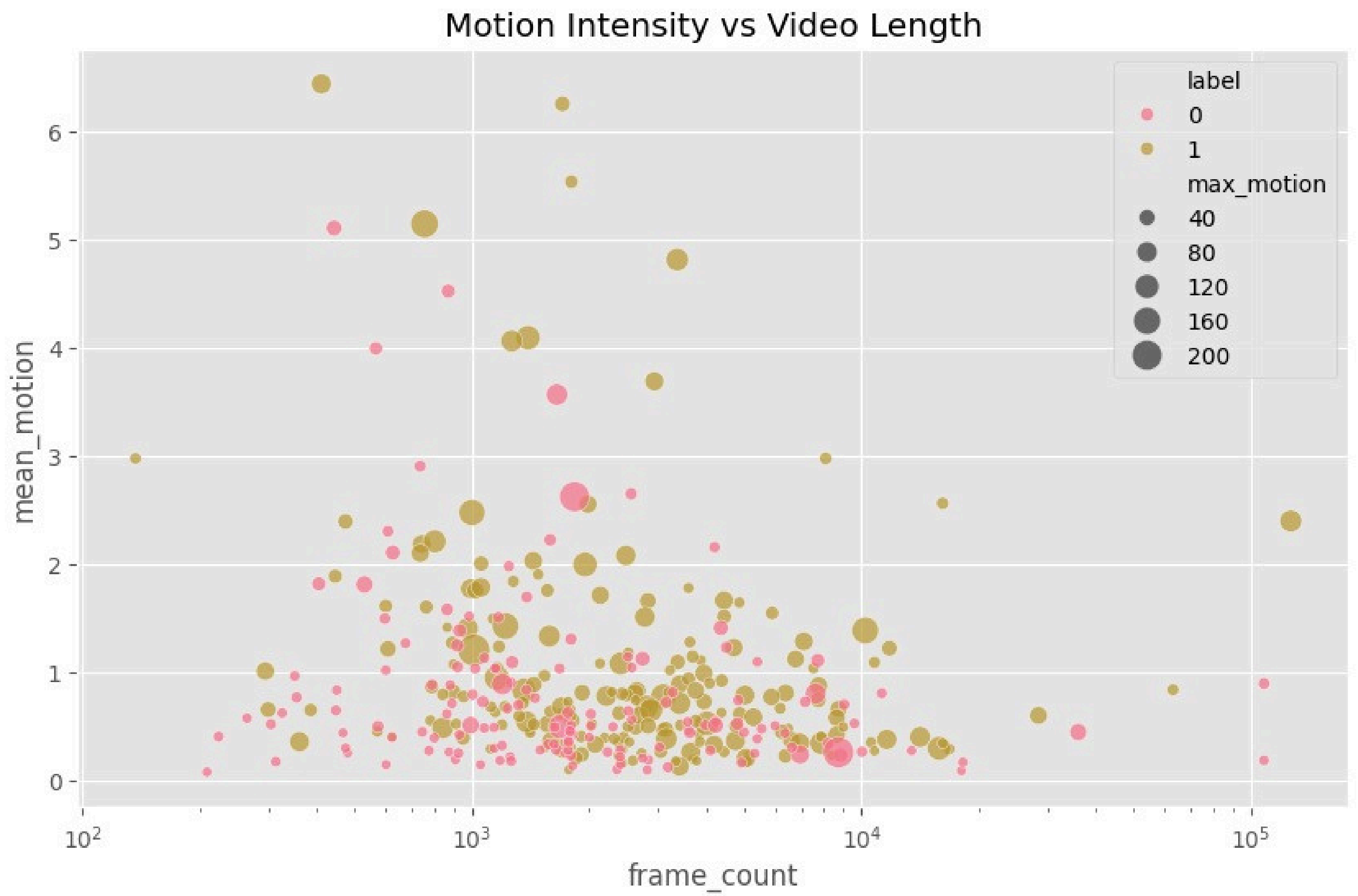
Motion Intensity vs Video Length (Scatter Plot)

Purpose

- To uncover correlations between video length and motion intensity, assessing whether crime events exhibit distinct patterns (e.g., short, intense Robberies vs. long, gradual Vandalism) for improved detection.

Insights Conveyed:

- The scatter plot conveys that anomalous videos often cluster at higher mean motion with larger sizes (high max motion), suggesting intense, brief crimes like Fights or Shootings (<1,500 frames). In contrast, normal videos form tighter clusters with lower, uniform motion, reflecting stable surveillance scenes.
- The size variation emphasizes peak motion as a crime signature, conveying that videos with large max motion (e.g., Explosions) are key anomaly candidates, regardless of length.



This scatter plot explores the relationship between frame count (x-axis, log-scaled) and mean motion (y-axis) in the dataset, with points colored by class (normal vs. anomalous) and sized by maximum motion. The logarithmic scale accommodates the dataset’s long video durations, while size adds a dimension for peak motion intensity.

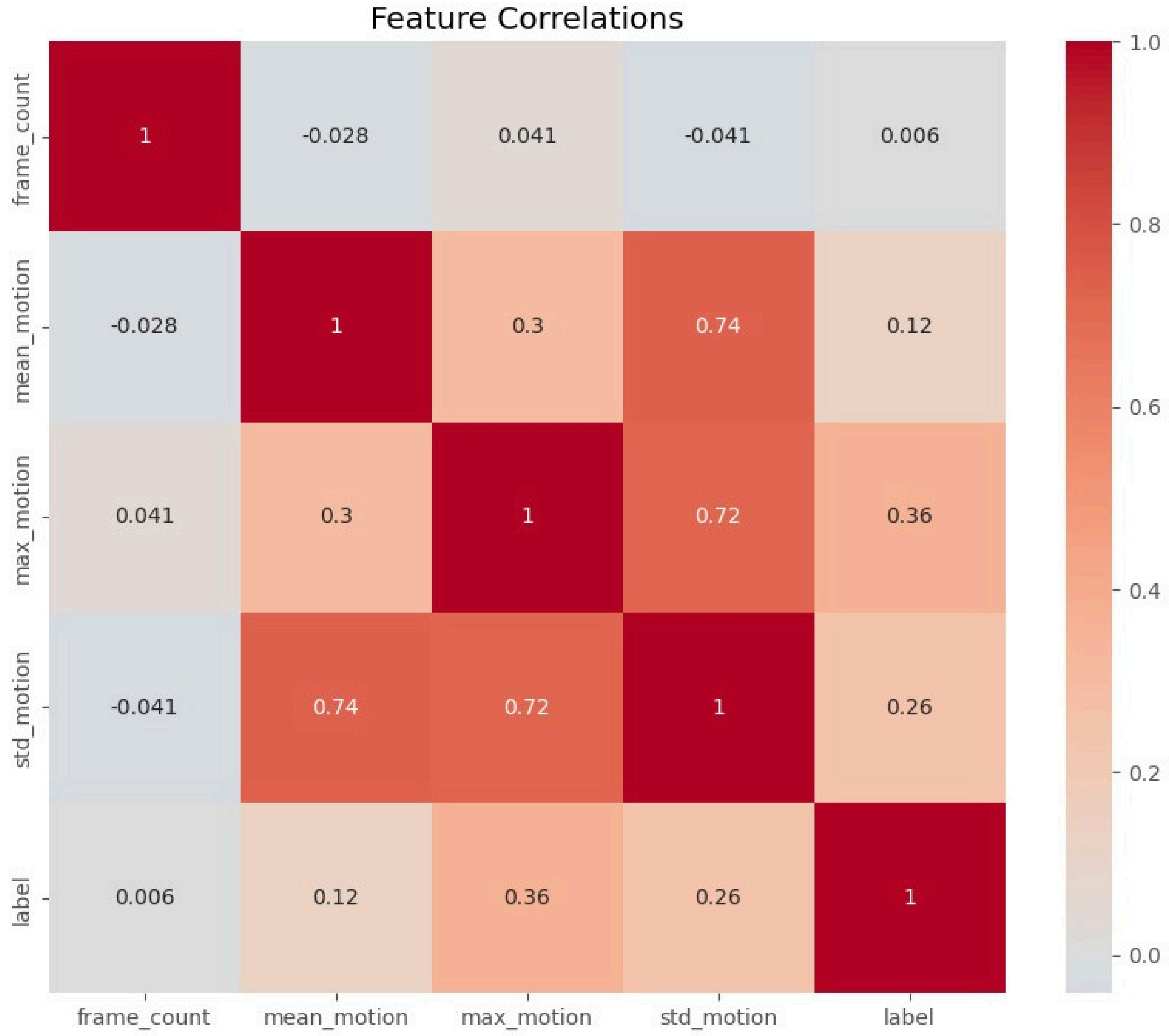
Feature Correlations (Heatmap)

Purpose

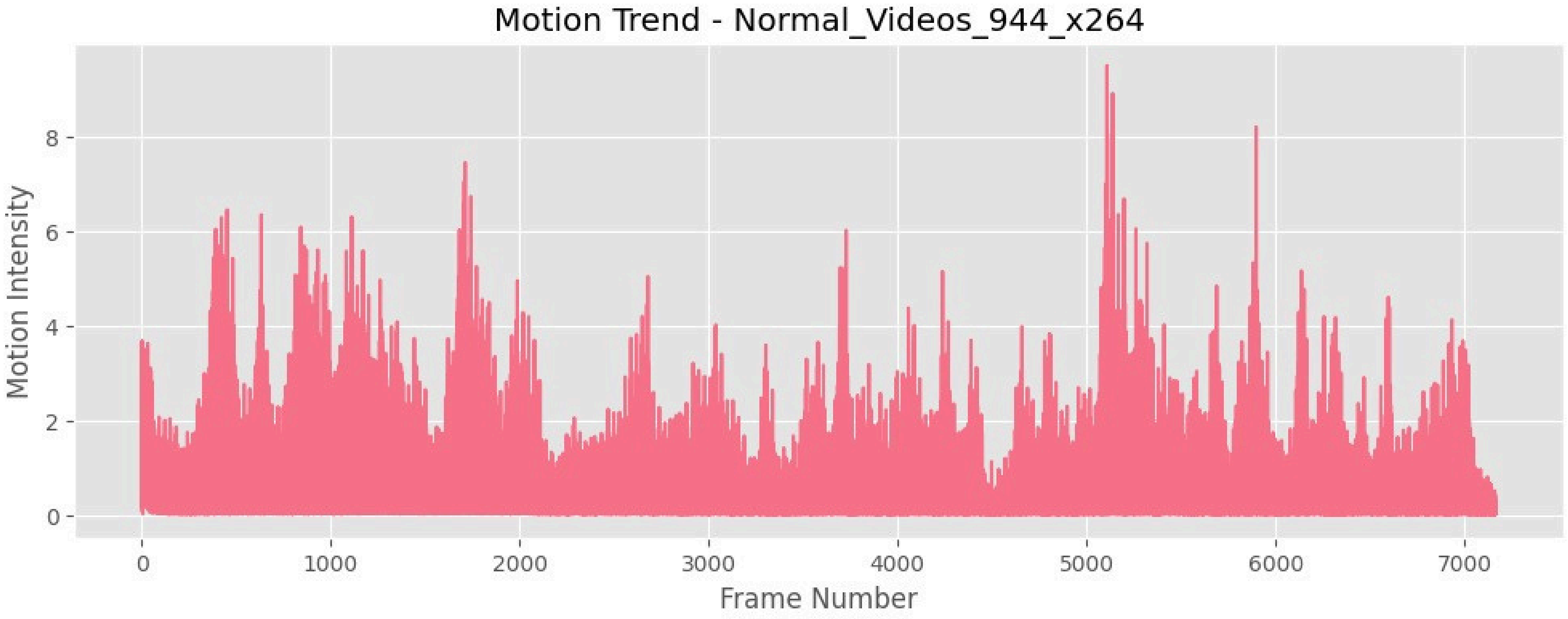
- To evaluate how well motion features and video length predict crime labels, identifying redundant or dominant features for optimizing detection algorithms.

Insights Conveyed & Message

- Label Correlations: The heatmap conveys that max motion and std. motion have moderate positive correlations with the anomaly label, signaling their effectiveness in detecting sudden (e.g., Assault) or irregular (e.g., Stealing) crimes. Mean motion and frame count show weaker links, suggesting limited standalone predictive power.
- Feature Intercorrelations: High correlations between motion features deliver a message of redundancy, implying that max motion might suffice, reducing computational load in models.
- Frame Count: The near-zero correlation with the label conveys that video length is not a strong crime indicator alone, urging focus on motion dynamics over duration.



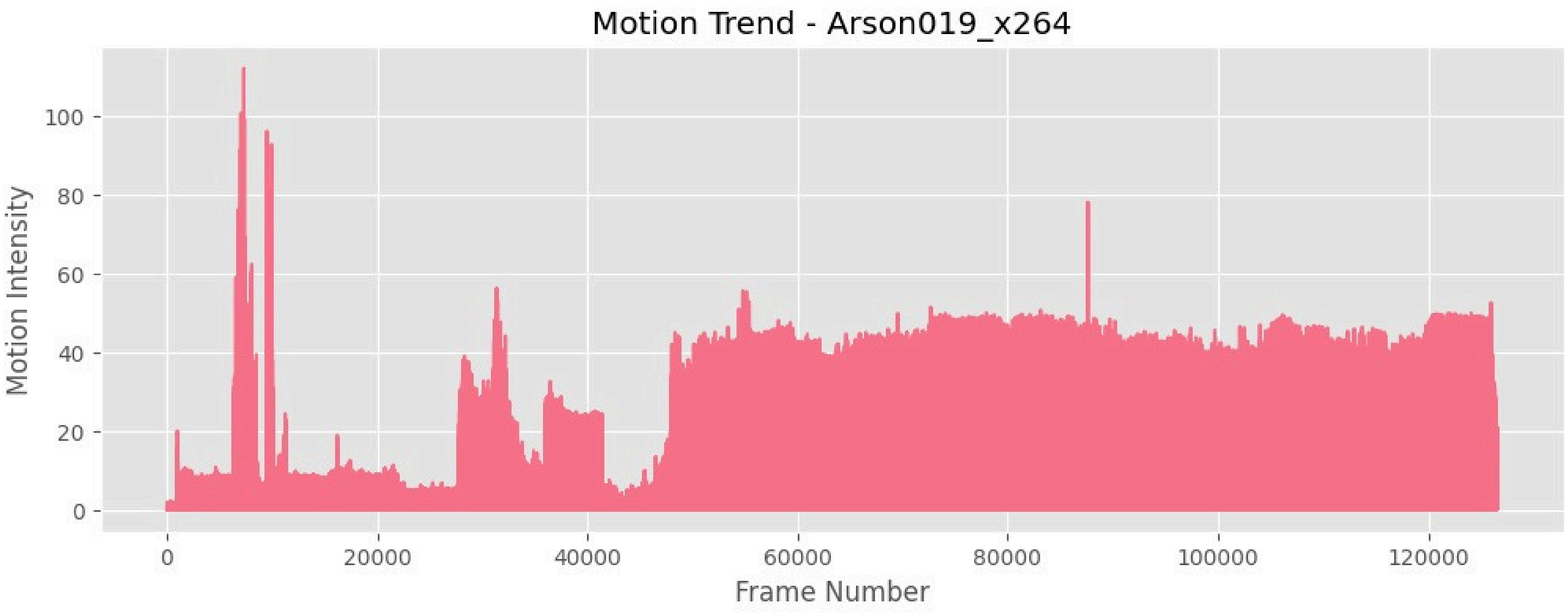
This heatmap displays correlation coefficients between frame count, mean motion, max motion, std motion, and the label in the UCF Crime dataset. Colors range from blue (negative) to red (positive), with annotated values indicating strength and direction of relationships.



Normal Video: Likely shows a stable or gradually varying trend, reflecting consistent activity (e.g., pedestrian flow).

Motion Patterns (Motion Trend)

Anomalous Video: Expected to exhibit sharp spikes or irregular patterns, indicating sudden or prolonged events.



Team members

• Rananjay Singh Chauhan	23BAI10080
• Anugya Chaubey	23BAI10550
• Tushar Kumar Tiwari	23BAI10551
• Nidhi Joshi	23BAI10572
• Abhay Chaudhary	23BAI10760
• Harsh Pandit	23BAI10772

THANK YOU!