An Invariant Backward Feature Analysis of Model-Based Malicious Activity Monitoring for Efficient Video Surveillance Using Deep Learning

K. Lokesh
Department of Computer Science and Engineering, School of Computing, College of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur, Chengalpattu, Tamil Nadu, India
klokesh280488@gmail.com

M. Baskar
Department of Computer Science and Engineering, School of Computing, College of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur, Chengalpattu, Tamil Nadu, India
baashkarinfo@gmail.com (corresponding author)

Received: 8 April 2025 | Revised: 9 May 2025, 1 August 2025, and 4 August 2025 | Accepted: 8 August 2025

 DOI: https://doi.org/10.48084/etasr.11370

ABSTRACT

The issue of malicious activity monitoring in video surveillance has been extensively studied, and several methods have been developed to address it. These approaches typically rely on attributes such as shapes, objects, textures, and sketches; however, their accuracy remains limited. To overcome these shortcomings, this paper presents an effective Convolutional Neural Network (CNN)-based malicious activity monitoring approach. The proposed technique detects harmful behavior by leveraging the invariant properties of drawings and their spatial positions across multiple preceding frames. To enhance input quality, Layer-Based Feature Normalization (LBFN) is applied to recorded video frames, removing noise and improving clarity. Feature segmentation is then performed using the Value-Oriented Segmentation (VOS) algorithm. The model maintains features extracted from the previous k frames and incorporates them to extract features from the current frame. Convolution and max-pooling layers are employed to convolve and normalize the extracted features. At the output layer, Sequential Position Support (SPS) and Sequential Sketch Support (SSS) are calculated using a variety of activity-related characteristics retained by the model and are iteratively evaluated across frame and feature sequences that the model has produced. Based on this process, the technique computes Malicious Activity Support (MAS) scores for different activity classes and assigns higher values to the most probable class. The proposed Invariant Backward Feature Analysis Model Convolutional Neural Network (IBFAM-CNN) achieves an accuracy of 97% in malicious activity monitoring and video surveillance.

Keywords-video surveillance; industrial security; malicious activity monitoring; sequential feature; invariant feature; Sequential Position Support (SPS); Sequential Sketch Support (SSS); Malicious Activity Support (MAS); Invariant Backward Feature Analysis Model Convolutional Neural Network (IBFAM-CNN)

I. INTRODUCTION

Security against malicious activity is a critical concern in all properties, with video surveillance serving as a primary method for monitoring suspicious behavior through real-time or recorded analysis of video feeds. However, manual supervision by human operators is limited and prone to oversight, necessitating automated video surveillance and malicious activity monitoring models.

Automated video surveillance employs various features and methodologies to achieve this goal. Some approaches detect activity using shape and object features, while others rely on sketch-based features. Traditional image-processing methods in the literature include Support Vector Machines (SVM), K-means clustering, and Particle Swarm Optimization (PSO). However, such machine learning techniques generally handle small datasets, limiting accuracy in high-complexity activity tracking. These limitations favor the adoption of deep learning methods such as Artificial Neural Networks (ANN), Convolutional Neural Networks (CNN), and Long Short-Term Memory (LSTM) networks.

Among deep learning approaches, CNNs are particularly effective because they convolve prominent features into reduced dimensions, enabling efficient processing of large datasets while improving classification accuracy. CNN architectures employ convolutional layers to reduce feature size and pooling layers to normalize features. Incorporating enforcement-based methods into malicious activity detection has demonstrated further improvement in accuracy. For example, in [1], Distorted Face Verification based on Surveillance Video Quality Analysis (DFV-SVQA) uses CNNs to recognize faces and evaluate surveillance video quality. Similarly, in [2], Generative Adversarial Network (GAN)-oriented texture synthesis approaches extracted dynamic texture content at the encoder stage, allowing classification through neuron-based correlation of spatial and temporal neighbors.

The choice of camera and system architecture also significantly influences surveillance performance. A Multi-Level Video Security (MuLViS) system [3], for instance, applies security ontology to enable automatic camera selection and restrict unauthorized access, while encrypting transmissions to prevent interception. Background reference features further enhance detection accuracy: a block-level Background Reference Frame (BRF) method [4] reduces redundancy by constructing reference frames from Surveillance Prediction Generative Adversarial Network (SPGAN) outputs and previous frames, with predictions informed by optical flow analysis. Other approaches, such as cloud-based surveillance systems [5], classify vulnerabilities, threats, and attacks using predefined taxonomies. Compression strategies [6] further improve frame quality by applying adaptive background updates and interpolating exchanged background data from nearby frames.

Additionally, advancements in the Internet of Things (IoT) and real-time surveillance have fostered several novel architectures. For example, the Internet of Video Things Video Surveillance System (IoVT-VSS) transmits video frames across IoT devices for efficient analysis [7], while Video Synthetic Aperture Radar (Video-SAR) models [8] produce sequential videos to enable continuous day-and-night monitoring. Scene-adaptive Octree-based models (SSOcT) [9] extract spatiotemporal structures for object classification, and trajectory-based surveillance methods [10] summarize video sequences to facilitate efficient analysis. Security can also be enhanced through encryption and steganographic techniques that safeguard data [11], and edge computing approaches that enable real-time failure detection [12].

Recent research emphasizes integrating multiple advanced methods to enhance surveillance robustness. For instance, the robust quadrangle algorithm [13] develops large-scale Distorted Surveillance Video Datasets (DSurVD) for pedestrian detection. Low-power coding techniques [14] segment input footage for efficient transmission, and frameworks like ViTrack [15] employ multi-video tracking with spatiotemporal classification. Moving target identification strategies [16], energy-efficient algorithms such as Simulated Annealing (SA) and JAYA [17], and MobileNet-based Faster Region-based CNN (R-CNN) detectors [18] have also improved real-time detection efficiency. Privacy-preserving approaches [19] combine object tracking with encryption, while landmark-free Conditional-GAN (CGAN) models [20] support reversible face de-identification. Sparsity-based regularization [21] enhances moving object detection, and hierarchical weighted fusion in CNN frameworks [22] improves classification reliability. Other methods include real-time segmentation and clustering [23], mobile edge computing for face recognition [24], geolocation-based feature coherence [25], and semantic region labeling using color, texture, and discrete cosine transform analysis [26]. Deep learning with attention mechanisms [27, 28] further improves the precision of suspicious activity detection. Meanwhile, Sketch and Size-oriented Malicious Activity Monitoring (SSMAM) models [29] enhance detection by segmenting frames and applying high-level intensity analysis, while modified threshold-centric K-means clustering [30] supports continuous classification. Multi-feature fusion methods [31] reduce distortion, improving recognition accuracy for complex activity detection.

These studies collectively show that the efficiency of video surveillance systems depends critically on both the volume and nature of extracted features. Most existing models, however, focus only on single-frame features in comparison to background frames, limiting their ability to accurately detect complex malicious activities. Such limitations are evident in actions that unfold over time; for example, detecting a slap requires analyzing motion across multiple frames (minimum 60 frames, assuming 12 frames per second). Accurate detection thus requires consideration of sequential frame features and backtracking of activity sketches, which existing methods neglect.

Addressing these gaps, this paper proposes an Invariant Backward Feature Analysis Model-based Malicious Activity Monitoring CNN (IBFAM-CNN) to improve detection accuracy. Layer-Based Feature Normalization (LBFN) is applied to remove noise and standardize frame data. Successive frame features are then retrieved to train the network, while Sequential Position Support (SPS) and Sequential Sketch Support (SSS) are measured at the output layer. These measures are iteratively evaluated, enabling calculation of Malicious Activity Support (MAS) for each activity class. The class with the highest MAS value is selected as the detected activity. By integrating sequential frame information, the IBFAM-CNN significantly enhances malicious activity monitoring, improving both accuracy and robustness in real-world surveillance scenarios.

II. METHODOLOGY

A. IBFAM-CNN for Malicious Activity Monitoring

The IBFAM-CNN model utilizes invariant sequential sketch and position features to detect and monitor malicious activities. It integrates LBFN to denoise video frames and Value-Oriented Segmentation (VOS) to cluster human-related features. The model extracts both location and sketch information from the current frame and a sequence of preceding k frames. These features are processed by convolutional and max-pooling layers to generate normalized representations for classification.

At the output layer, SPS and SSS are measured for the extracted features. These measures are iteratively refined over successive frames to calculate MAS for different activity classes. The activity class with the highest MAS value is selected as the detected malicious activity. The functional workflow of IBFAM-CNN is illustrated in Figure 1.

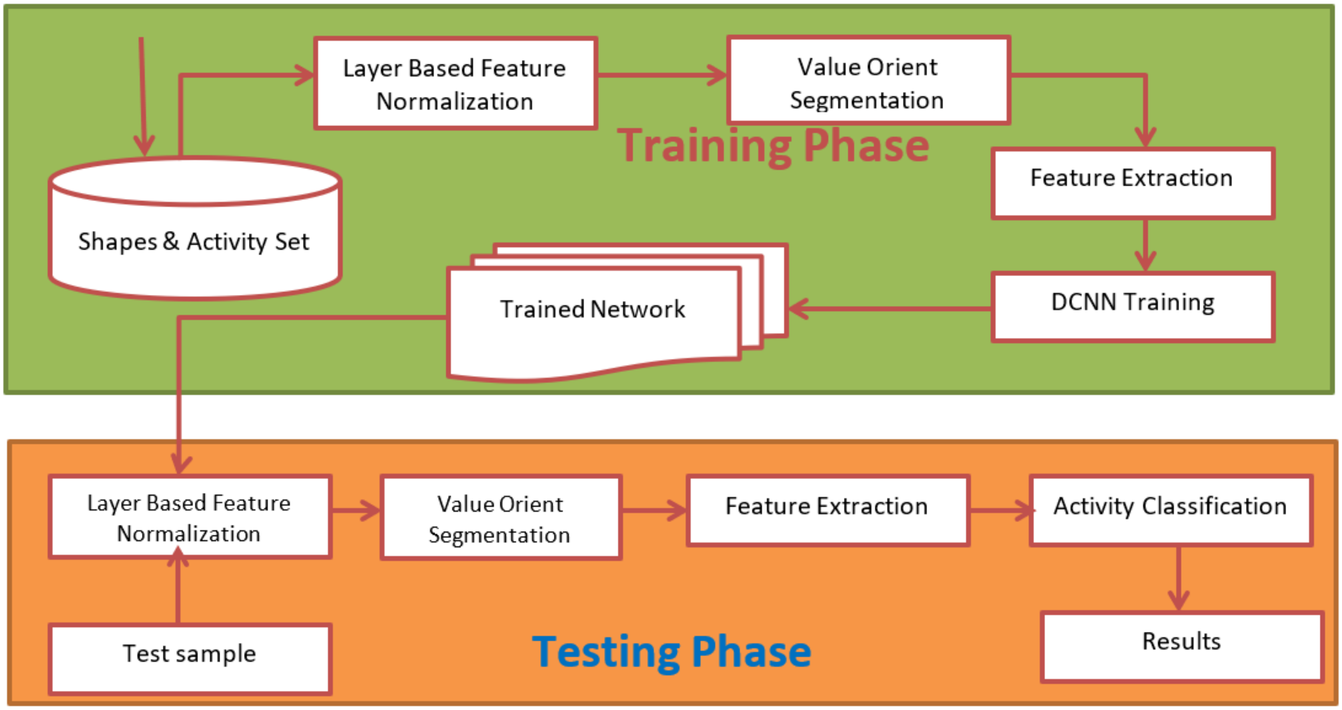

Fig. 1. Functional workflow of IBFAM-CNN scheme.

The upper portion depicts the training process, which involves feature normalization and segmentation to group relevant data, followed by model training on the extracted features. The lower portion illustrates the testing phase, where normalization, segmentation, and feature extraction are applied to incoming frames.

B. *Layer-Based Feature Normalization (LBFN)*

LBFN preserves object and human-related details in video frames, enhancing detection accuracy. The model normalizes each image layer independently using a sliding window approach, extracting features to compute a Regional Retention Value (RRV), defined as the minimum pixel intensity relative to the median intensity of the window. Based on RRV, pixel intensities are adjusted to improve feature representation for segmentation.

```
Algorithm:
Given: Video Frame Vf
Obtain: Normalized Image Ning
Start
Read Vf
Initialize window size ws = 5
At each layer l
For each window w
```
Pixel set Ps = $\sum pixels \in w$

Compute Median Pm = $\sum_{i=1}^{size(Ps)} Ps(i).value \Big/ size(Ps)$

Compute regional retention value rrv =

$$\frac{\frac{\sum_{i=1}^{size(Ps)} Ps(i).value<Pm}{size(Simgs(n))}}{\underset{i=1}{Count}(Simgs(i).coordinate==O(j).coordinate)} - Pm$$

```
For each pixel in w
If pi.value<Pm &&pi.value>rrv then
Pi.value = Pm
End
End
End
End
```
N_{img} = vf

Stop

C. *Value-Oriented Segmentation (VOS)*

VOS clusters normalized image data based on pixel intensity distributions, isolating distinct objects such as human figures. First, the normalized image N_{img} is converted to grayscale, and histograms are computed for defined regions. Pixels above the median intensity are counted, and the three most frequent pixel values are identified to form peak and median sets. These sets serve as references for clustering pixels into objects.

Algorithm:

Given: Normalized_Image N_{img}

Obtain: Segmented_image set S_{imgs}

Start
Fetch N_{img}
Initiate count set cs, peak set Ps, median set ms
Initiate window size w.
At any region R
Crop region image $Ri = \int Crop(nimg, R)$
Hist H = generate histogram (Ri)
Median $me = \sum_{i=1}^{Size(Ri)} Ri(i).value \Big/ Size(Ri)$
Identify the peak value set $pvs = \sum_{i=1}^{size(Ri)} Ri(i).value > me \;\&\&\; Ri(i).value.Occurrence \geq MaxThree(H)$
Add pvs to $ps = (\sum peakvalues \in Ps) \cup pvs$
Add me to the median set $ms = (\sum median \in ms) \cup me$
End
Initialize object set Os = size(pvs)
For each pixel p in N_{img}
For each object o
If Dist(me(o),p.value)<th then
Add pixel p to object o.
End
End
End
For each object o
Produce segmented image S_{imgs}.
Add to $S_{imgs} = (\sum img \in S_{imgs}) \cup S_{imgs}$
End
Stop

By comparing pixel intensities to peak and median values, this method effectively segments the image into objects, even in complex scenes. For example, considering a 5-pixel region *R* with a histogram of 256 values, the method identifies peak intensity values and corresponding medians to create sets used for segmentation across the image.

D. Feature Extraction

From the segmented image set S_{imgs}, the feature extraction process derives both positional and sketch features. Each segmented image is matched to an Object Dictionary (OD) to estimate a Human Sketch Score (HSS) for various object classes, thereby identifying relevant human-related sketch and positional features. HSS determines which objects should be treated as human features. The extracted sketch and position attributes are combined to form a feature vector used for training and testing the model.

Algorithm:
Given: Object Dictionary OD, Segmented image set S_{imgs}
Obtain: Feature vector Fv
Start
Read OD and S_{imgs}.
For each image in S_{imgs}:
For each object class oc:
For each object o:
Estimate Human Sketch score (HSS) = $\frac{\underset{i=1}{\overset{size(Simgs(n))}{Count(Simgs(i).coordinate == O(j).coordinate)}}}{size(Simgs(n))}$
End
Estimate Cumulative HSS as CHSS = $\frac{\sum HSS}{size(Oc)}$
End
End
Human image Himg = Choose the class with maximum CHSS.
Feature vector Fv = {Extract sketch, Position from Himg}
Stop

E. Deep Convolutional Neural Network (DCNN) Training

The proposed Deep Convolutional Neural Network (DCNN) comprises two convolutional layers and pooling layers. The first convolution layer reduces extracted features to 250 dimensions, preserving sketch-related attributes. The second convolution layer counts pixels in each image quadrant to extract positional features, which are then converted into a one-dimensional feature vector. Pooling layers normalize these values, enhancing the stability of training. The DCNN is trained using extracted sketch and position features. Neurons in the network are initialized with these features to compute SPS and SSS, which are combined to measure MAS for activity classification.

Algorithm:
Given: Activity Data set SADs
Obtain: DCNN
Start
Read SADs
Initialize DCNN.
For each image vimg
Primg = Perform Layer-Based Feature Normalization (vimg)
Seimgs = Perform Value-oriented segmentation (primg)
[Sketch, position] = Feature Extraction (S_{imgs})
Add sketch and position set Skps.
Generate neuron N = Initialize with Skps.
End
Stop

F. Activity Classification

Activity classification uses spatial and sketch features extracted from a sequence of frames. The procedure begins with LBFN to remove noise and enhance frame quality, followed by VOS to isolate objects of interest. Extracted human features are passed through the trained DCNN, which calculates SPS and SSS values for the current and previous frames. These values are iteratively combined to compute MAS for each activity class.

SPS uses positional continuity across multiple frames to detect malicious movement patterns. For example, detecting a "slapping" action requires tracking hand motion over several frames. Similarly, SKS captures sketch changes over time to distinguish different activities.

```
Algorithm:
Given: DCNN, Test sample Ts
Obtain: Class C
Start
Fetch DCNN and Ts
Primg = Level based normalization (Ts)
Seimg = Apply value orient segmentation
(primg)
[Sketch, Position] = Feature Extraction
(seimg)
[Sketch set Skes, position set pos] =
Collect sketch and position features from
previous Frames
Add sketch, position to skes and pos.
Pass sks and pos through the DCNN.
For each class C
For each feature in skes
For each layer l
For each neuron n
Compute Sequential position support Sps.
```

Sps= $\frac{\sum_{i=1}^{size(Skes(k))} Sks.Feature \in N.PositionFeatures}{size(Skes(k))}$

```
End
Compute Sequential Sketch Support.
SSS =
```

$\frac{\sum_{i=1}^{size(Skes(i).Sketch)} Count(N.Skes.value(i) == Skes(i).Sketch.value(i))}{4}$

```
End
```

Compute MAS = $\frac{\sum SKS}{Size(Class)} \times \frac{\sum SPS}{Size(Class)}$

```
End
End
Class = Elect maximum MAS valued class.
Stop
```

This process yields the activity classification by identifying the class with the highest MAS value, enabling robust detection of malicious actions.

III. RESULTS AND DISCUSSION

The proposed IBFAM-CNN was implemented in MATLAB and evaluated under controlled experimental conditions. Experiments were conducted on an Ubuntu system with 16 GB Random Access Memory (RAM) and an NVIDIA Tesla P100 Graphics Processing Unit (GPU). Evaluation was performed using the DCSASS dataset available on Kaggle [32]. This dataset contains videos categorized into 15 classes of activities: Slapping, Kicking, Abuse, Arrest, Arson, Assault, Accident, Burglary, Explosion, Fighting, Robbery, Shooting, Stealing, Shoplifting, and Vandalism. Each video is labeled as normal (0) or abnormal (1). The dataset comprises 16,853 videos: 9,676 labeled as normal and 7,177 as abnormal.

Table 1 shows performance evaluation constraints. For each activity class, approximately 1 million images were extracted for training. Videos from 300 participants were utilized for training and testing.

A. *Detection Accuracy*

Table II compares the detection accuracy of IBFAM-CNN against BRF, SPGAN, and DSurVD across varying numbers of activities (5, 10, and 15). IBFAM-CNN consistently achieves the highest accuracy, improving from 83% for 5 activities to 97% for 15 activities. SPGAN performs competitively (77%-86%), while BRF and DSurVD show moderate performance (73%-82% and 72%-81%, respectively). Detection accuracy improves across all models as activity complexity increases, but IBFAM-CNN shows the most substantial gain, highlighting its robustness in real-world malicious activity detection.

TABLE I. EVALUATION DETAILS

Parameter	Value
Total Activities	15
Total Images	15 million
Tool Used	MATLAB
Number of Users	300

TABLE II. PERFORMANCE IN DETECTION OF MALICIOUS ACTIVITY

Malicious Activity Detection Accuracy % vs Number of Activities			
Activities	5 Activities	10 Activities	15 Activities
IBFAM-CNN	83	89	97
BRF	73	77	82
SPGAN	77	82	86
DSurVD	72	76	81

Figure 2 illustrates detection accuracy trends, showing that IBFAM-CNN outperforms all compared models, particularly at higher activity levels.

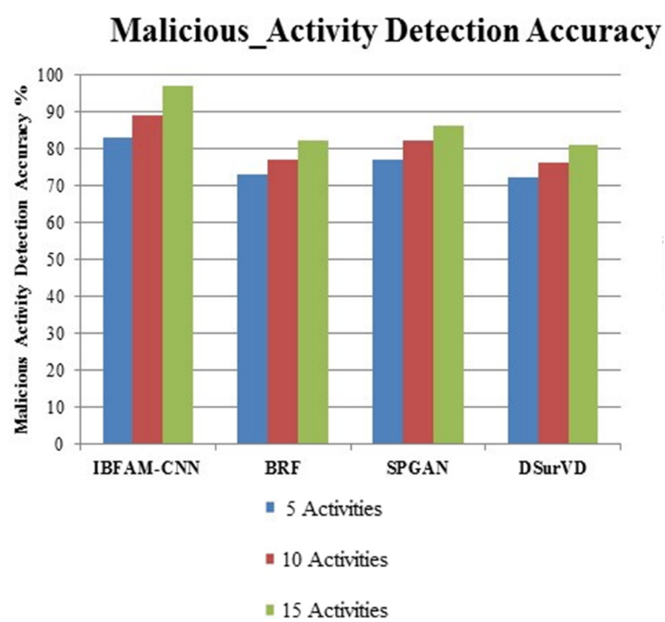

Fig. 2. Accuracy in malicious activity detection.

B. False Detection Ratio

The inefficiency in classifying malicious activity is gauged as a false ratio in Table III across the four evaluated models. IBFAM-CNN achieves the lowest false detection rate, reducing from 17% for 5 activities to 3% for 15 activities, demonstrating strong reliability and precision. BRF records the highest false rate (27% → 18%), followed by SPGAN (23% → 14%) and DSurVD (18% → 19%). These results indicate IBFAM-CNN's superior ability to minimize erroneous classification as activity complexity increases.

TABLE III. FALSE RATE IN MALICIOUS ACTIVITY DETECTION

False Ratio % vs Number of Activities			
Activities	5 Activities	10 Activities	15 Activities
IBFAM-CNN	17	11	3
BRF	27	23	18
SPGAN	23	18	14
DSurVD	18	14	19

Figure 3 visualizes false detection trends, reaffirming IBFAM-CNN's ability to consistently reduce false positives compared to alternative methods.

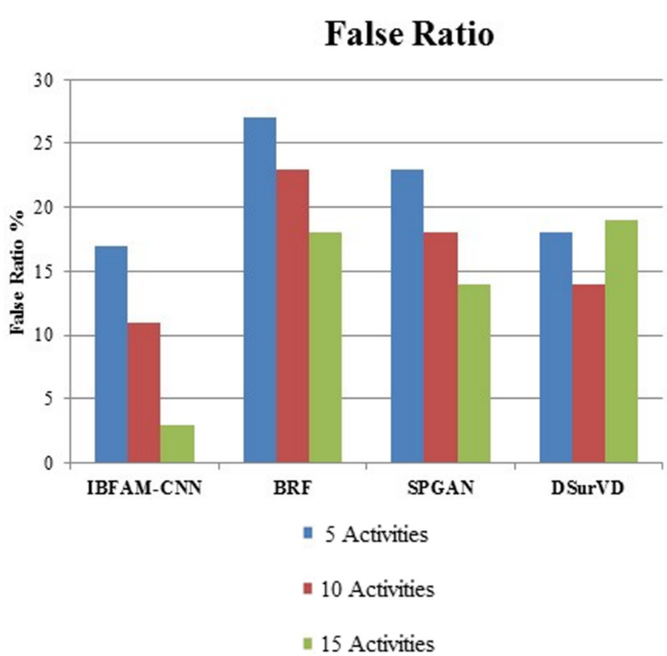

Fig. 3. False ratio in malicious activity detection.

C. Time Complexity

Table IV compares the time complexity for activity classification across models. The most time-efficient model was IBFAM-CNN, with classification time increasing moderately from 21 seconds for 5 activities to 45 seconds for 15 activities. In contrast, BRF exhibits the highest time complexity (67→89 seconds), while SPGAN (56→81 seconds) and DSurVD (49→75 seconds) show higher computational costs than IBFAM-CNN.

TABLE IV. TIME COMPLEXITY IN MALICIOUS ACTIVITY DETECTION

Time Complexity (s) vs Number of Activities			
Activities	5 Activities	10 Activities	15 Activities
IBFAM-CNN	21	32	45
BRF	67	79	89
SPGAN	56	71	81
DSurVD	49	63	75

Figure 4 illustrates the efficiency advantage of IBFAM-CNN, making it the most suitable choice for large-scale or real-time malicious activity detection scenarios.

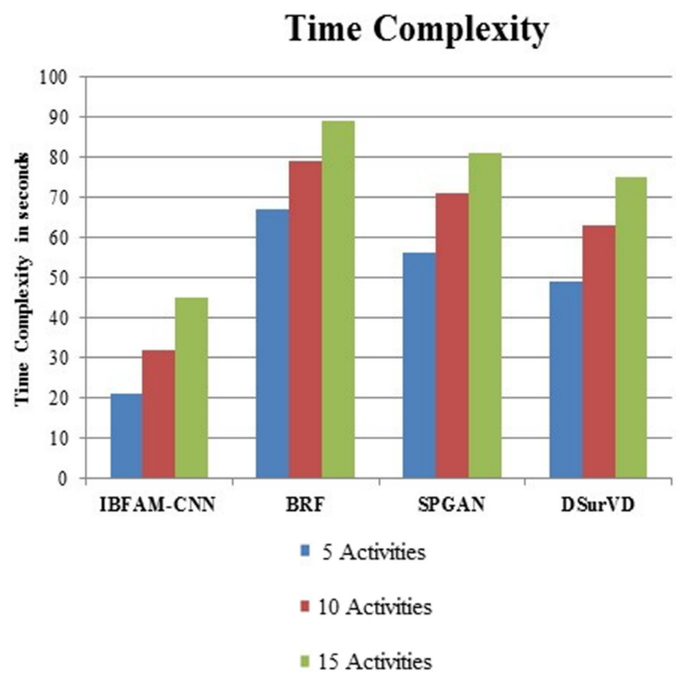

Fig. 4. Analysis time complexity in malicious activity detection.

D. Summary of Findings

Overall, IBFAM-CNN achieves 97% classification accuracy on the DCSASS dataset across diverse activity classes, significantly outperforming comparative models in detection accuracy, false detection ratio, and computational efficiency. These findings highlight IBFAM-CNN's scalability, robustness, and suitability for real-world video surveillance applications where both precision and efficiency are critical.

IV. CONCLUSION

This study presents the Invariant Backward Feature Analysis Model Convolutional Neural Network (IBFAM-CNN) for malicious activity monitoring in video surveillance. The model leverages invariant sequential sketch and positional attributes to enhance detection accuracy. Preprocessing is achieved using the Layer-Based Feature Normalization (LBFN)technique, followed by Value-Oriented Segmentation (VOS) to group human features. From the segmented images, positional and sketch features are extracted across sequences of frames to compute various support measures. By incorporating sequential frame features and applying backward tracking, IBFAM-CNN significantly improves the precision of malicious activity detection. Unlike conventional methods that consider only individual frames, this model integrates features from multiple preceding frames, achieving enhanced robustness and achieving up to 97% detection accuracy in identifying harmful activities.

REFERENCES

[1] W. Heng, T. Jiang, and W. Gao, "How to Assess the Quality of Compressed Surveillance Videos Using Face Recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 8, pp. 2229–2243, Aug. 2019, https://doi.org/10.1109/TCSVT.2018.2866701.

[2] K. Yang, D. Liu, Z. Chen, F. Wu, and W. Li, "Spatiotemporal Generative Adversarial Network-Based Dynamic Texture Synthesis for Surveillance Video Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 1, pp. 359–373, Jan. 2022, https://doi.org/10.1109/TCSVT.2021.3061153.

[3] A. Shifa *et al.*, "MuLViS: Multi-Level Encryption Based Security System for Surveillance Videos," *IEEE Access*, vol. 8, pp. 177131–177155, 2020, https://doi.org/10.1109/ACCESS.2020.3024926.

[4] L. Zhao, S. Wang, S. Wang, Y. Ye, S. Ma, and W. Gao, "Enhanced Surveillance Video Compression With Dual Reference Frames Generation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 3, pp. 1592–1606, Mar. 2022, https://doi.org/10.1109/TCSVT.2021.3073114.

[5] D. Aklamati, B. Abdus-Shakur, and T. Kacem, "Security Analysis of AWS-based Video Surveillance Systems," in *2021 International Conference on Engineering and Emerging Technologies (ICEET)*, Istanbul, Turkey, Oct. 2021, pp. 1–6, https://doi.org/10.1109/ICEET53442.2021.9659574.

[6] L. Wu, K. Huang, H. Shen, and L. Gao, "Foreground-Background Parallel Compression With Residual Encoding for Surveillance Video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 7, pp. 2711–2724, Jul. 2021, https://doi.org/10.1109/TCSVT.2020.3027741.

[7] T. Sultana and K. A. Wahid, "Choice of Application Layer Protocols for Next Generation Video Surveillance Using Internet of Video Things," *IEEE Access*, vol. 7, pp. 41607–41624, 2019, https://doi.org/10.1109/ACCESS.2019.2907525.

[8] M. R. Khosravi and S. Samadi, "Mobile multimedia computing in cyber-physical surveillance services through UAV-borne Video-SAR: A taxonomy of intelligent data processing for IoMT-enabled radar sensor networks," *Tsinghua Science and Technology*, vol. 27, no. 2, pp. 288–302, Apr. 2022, https://doi.org/10.26599/TST.2021.9010013.

[9] Y. Yang, H. Kim, H. Choi, S. Chae, and I.-J. Kim, "Scene Adaptive Online Surveillance Video Synopsis via Dynamic Tube Rearrangement Using Octree," *IEEE Transactions on Image Processing*, vol. 30, pp. 8318–8331, 2021, https://doi.org/10.1109/TIP.2021.3114986.

[10] S. A. Ahmed, D. P. Dogra, S. Kar, and P. P. Roy, "Trajectory-Based Surveillance Analysis: A Survey," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 7, pp. 1985–1997, Jul. 2019, https://doi.org/10.1109/TCSVT.2018.2857489.

[11] N. Kanwal *et al.*, "Preserving Chain-of-Evidence in Surveillance Videos for Authentication and Trust-Enabled Sharing," *IEEE Access*, vol. 8, pp. 153413–153424, 2020, https://doi.org/10.1109/ACCESS.2020.3016211.

[12] H. Sun, W. Shi, X. Liang, and Y. Yu, "VU: Edge Computing-Enabled Video Usefulness Detection and its Application in Large-Scale Video Surveillance Systems," *IEEE Internet of Things Journal*, vol. 7, no. 2, pp. 800–817, Feb. 2020, https://doi.org/10.1109/JIOT.2019.2936504.

[13] Y. Fang, G. Ding, Y. Yuan, W. Lin, and H. Liu, "Robustness Analysis of Pedestrian Detectors for Surveillance," *IEEE Access*, vol. 6, pp. 28890–28902, 2018, https://doi.org/10.1109/ACCESS.2018.2840329.

[14] H. Kim and H.-J. Lee, "A low-power surveillance video coding system with early background subtraction and adaptive frame memory compression," *IEEE Transactions on Consumer Electronics*, vol. 63, no. 4, pp. 359–367, Nov. 2017, https://doi.org/10.1109/TCE.2017.015073.

[15] L. Cheng, J. Wang, and Y. Li, "ViTrack: Efficient Tracking on the Edge for Commodity Video Surveillance Systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 3, pp. 723–735, Mar. 2022, https://doi.org/10.1109/TPDS.2021.3081254.

[16] J. Huang, A. Huang, and L. Wang, "Intelligent Video Surveillance of Tourist Attractions Based on Virtual Reality Technology," *IEEE Access*, vol. 8, pp. 159220–159233, 2020, https://doi.org/10.1109/ACCESS.2020.3020637.

[17] S. Ghatak, S. Rup, B. Majhi, and M. N. S. Swamy, "HSAJAYA: An Improved Optimization Scheme for Consumer Surveillance Video Synopsis Generation," *IEEE Transactions on Consumer Electronics*, vol. 66, no. 2, pp. 144–152, May 2020, https://doi.org/10.1109/TCE.2020.2981829.

[18] W. Liu, S. Liao, and W. Hu, "Perceiving Motion From Dynamic Memory for Vehicle Detection in Surveillance Videos," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 12, pp. 3558–3567, Dec. 2019, https://doi.org/10.1109/TCSVT.2019.2906195.

[19] X. Tian, P. Zheng, and J. Huang, "Robust Privacy-Preserving Motion Detection and Object Tracking in Encrypted Streaming Video," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 5381–5396, 2021, https://doi.org/10.1109/TIFS.2021.3128817.

[20] H. Proenca, "The UU-Net: Reversible Face De-Identification for Visual Surveillance Video Footage," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 2, pp. 496–509, Feb. 2022, https://doi.org/10.1109/TCSVT.2021.3066054.

[21] B.-H. Chen, L.-F. Shi, and X. Ke, "A Robust Moving Object Detection in Multi-Scenario Big Data for Video Surveillance," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 4, pp. 982–995, Apr. 2019, https://doi.org/10.1109/TCSVT.2018.2828606.

[22] K. Muhammad, T. Hussain, M. Tanveer, G. Sannino, and V. H. C. De Albuquerque, "Cost-Effective Video Summarization Using Deep CNN With Hierarchical Weighted Fusion for IoT Surveillance Networks," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4455–4463, May 2020, https://doi.org/10.1109/JIOT.2019.2950469.

[23] L. Zhao, Z. He, W. Cao, and D. Zhao, "Real-Time Moving Object Segmentation and Classification From HEVC Compressed Surveillance Video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 6, pp. 1346–1357, Jun. 2018, https://doi.org/10.1109/TCSVT.2016.2645616.

[24] H. Hu *et al.*, "Video Surveillance on Mobile Edge Networks—A Reinforcement-Learning-Based Approach," *IEEE Internet of Things Journal*, vol. 7, no. 6, pp. 4746–4760, Jun. 2020, https://doi.org/10.1109/JIOT.2020.2968941.

[25] E. Taghavi *et al.*, "Geo-registration and Geo-location Using Two Airborne Video Sensors," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 4, pp. 2910–2921, Aug. 2020, https://doi.org/10.1109/TAES.2020.2995439.

[26] H. Huang, A. V. Savkin, and W. Ni, "Online UAV Trajectory Planning for Covert Video Surveillance of Mobile Targets," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 2, pp. 735–746, Apr. 2022, https://doi.org/10.1109/TASE.2021.3062810.

[27] R. Radhika and A. Muthukumaravel, "Video Surveillance and Deep Learning Enhancing Security through Suspicious Activity Detection," in *2024 International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS)*, Hassan, India, Aug. 2024, pp. 1–6, https://doi.org/10.1109/IACIS61494.2024.10721938.

[28] M. Pangavhane *et al.*, "Real-Time Deep Learning-Driven Surveillance with Spatiotemporal Feature Extraction for Detection of Anomalous Human Behavior Across Dynamic Environments," *International Journal of Safety and Security Engineering*, vol. 15, no. 1, pp. 105–111, Jan. 2025, https://doi.org/10.18280/ijsse.150112.

[29] K. Lokesh and M. Baskar, "Sketch and Size Orient Malicious Activity Monitoring for Efficient Video Surveillance Using CNN," *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 8, 2024, https://doi.org/10.14569/IJACSA.2024.0150831.

[30] L. Shana and C. Seldev Christopher, "A deep learning behavior analysis model for efficient video surveillance using multi pose features," *Ain Shams Engineering Journal*, vol. 16, no. 2, Feb. 2025, Art. no. 103245, https://doi.org/10.1016/j.asej.2024.103245.

[31] P. Nuthakki *et al.*, "Deep Learning based Multilingual Speech Synthesis using Multi Feature Fusion Methods," *ACM Transactions on Asian and Low-Resource Language Information Processing*, Sep. 2023, Art. no. 3618110, https://doi.org/10.1145/3618110.

[32] W. Sultani, C. Chen, and M. Shah, "Real-World Anomaly Detection in Surveillance Videos," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, Jun. 2018, pp. 6479–6488, https://doi.org/10.1109/CVPR.2018.00678.