INTERNATIONAL JOURNAL OF ELECTRONICS AND COMPUTER APPLICATIONS

ORIGINAL ARTICLE

Article access online

OPEN ACCESS

Received: 25.01.2025

Accepted: 10.05.2025

Published: 28.06.2025

Citation: Patil A. (2025). Anomaly Detection in Videos Using Deep Learning. International Journal of Electronics and Computer Applications. 2(1): 90-96. https://doi.org/10.70968/ijeaca.v2i1.D1004

* **Corresponding author**.

Funding: None

Competing Interests: None

ISSN
Print: XXXX-XXXX
Electronic: 3048-8257

Anomaly Detection in Videos Using Deep Learning

Ashwini Patil[1]*

1 Assistant Professor, Department of E&TC, Ajeenkya D Y Patil School of Engineering, Pune, Maharashtra, India

Abstract

In recent decades, surveillance cameras have been widely deployed in various locations for security and monitoring. The video data analysis captured by these cameras plays a crucial role in event prediction, real-time tracking, and goal-driven applications such as anomaly and intrusion detection. With advancements in Artificial Intelligence (AI), deep learning-based approaches, particularly Convolutional Neural Networks (CNNs), have significantly improved anomaly detection accuracy. This study proposes a novel deep-learning framework for detecting anomalies in video surveillance, leveraging CNNs for spatial feature extraction. In recent decades, surveillance cameras have been widely deployed in various locations for security and monitoring. The video data analysis captured by these cameras plays a crucial role in event prediction, real-time tracking, and goal-driven applications such as anomaly and intrusion detection. With advancements in Artificial Intelligence (AI), deep learning-based approaches, particularly Convolutional Neural Networks (CNNs), have significantly improved anomaly detection accuracy. This study proposes a novel deep-learning framework for detecting anomalies in video surveillance, leveraging CNNs for spatial feature extraction. The approach is inspired by previous studies, such as "Anomaly Detection in Surveillance Videos Using Deep Learning"[(1)], which demonstrated high efficiency in recognizing abnormal patterns. The UCSD dataset has been used to assess the suggested approach, showing improved accuracy in anomaly detection compared to existing methods. The findings highlight the potential of deep learning in enhancing automated surveillance systems, contributing to intelligent security monitoring and public safety.[(1)]

Keywords: Deep Learning; Anomaly Detection; Surveillance Camera

Introduction

Unsupervised machine learning techniques for anomaly detection remain a widely debated topic in the field of machine learning. Identifying anomalies by learning from normal data has important and varied applications[(2)], and anomaly detection is highly dependent

on the environment, context, and particular anomaly scenario [3,4]. Anomalies are defined as odd, irregular, unexpected, and unpredictable events or behaviors that differ from established patterns [5].

The characteristics of aberrations differ across diverse scenarios. Current supervised outlier detection techniques, including CNN-driven strategies, depend on annotated datasets, which can be difficult to acquire due to the high-dimensional characteristics of video information. The complexity of video data impacts both representation and model development [6]. This study focuses on anomaly detection in surveillance camera footage, where the challenge is greater compared to other data types, as it involves both detection methods and video processing techniques [7].

Analyzing footage from surveillance cameras in densely populated areas is challenging, especially when done in real-time, as it adds to the complexity. One of the most efficient ways to process this data and identify relevant patterns is by leveraging advanced AI techniques like deep learning. These methods are particularly useful for handling large-scale data due to their ability to function as complete, automated systems. Such systems eliminate the need for manual feature selection [7]. The primary goal of deep learning is to extract valuable information from high-dimensional, complicated data [8].

This research introduces a deep learning-driven technique for detecting irregularities. The framework comprises two fundamental phases: a training module and a recognition classifier. The initial phase emphasizes feature extraction through a five-layered deep structure. The subsequent phase is dedicated to identifying deviations and integrates five advanced neural network classifiers alongside a restoration model. Each unit within the recognition phase produces a categorized outcome and an associated metric. Ultimately, a collective classifier merges these outputs to determine the conclusive detection result.

The primary contribution of this study is the deployment of advanced neural network techniques at every phase of irregularity recognition. In the following segments, a summary and fundamental principles of detecting anomalies in video data through deep learning strategies are first presented. Section II covers prior research in the field, while Section III elaborates on the newly proposed technique in detail. Lastly, the final section presents assessments to highlight the enhancements and benefits of the suggested method compared to existing approaches.

2 Background

Owing to the presence of abundant and insightful details in recordings and their convenient availability, academic investigators have shown interest in examining and handling this type of information. A challenge in visual data interpretation is recognizing items in scenes from video [9].

Moreover, detecting unusual patterns in videos has been a widely debated subject of study in recent years. Over the past few years, neural network-based techniques have also been introduced for implementing irregularity identification strategies. In all anomaly recognition methodologies, training is performed exclusively using standard data. Another crucial aspect concerning deviations is that atypical occurrences are generally infrequent events that take place significantly less often than other routine happenings [5].

The difficulties in identifying irregularities in footage include processing speed, real-time alerts, and pinpointing their location. It is important to note that locating anomalies is essential, yet most current systems and datasets do not provide this feature certain techniques conduct region identification during initial processing, typically by analyzing and contrasting video frames, which contributes to enhancing precision [10,11]. In other terms, the majority of prevailing methodologies and datasets merely recognize the existence of irregularities without pinpointing their precise location [12]. Furthermore, contemporary strategies suffer from inadequate training samples and ambiguous anomaly classifications, while the substantial expense of attribute extraction directly influences detection efficiency [6].

A popular technique for finding anomalies is a dual-class classifier that has two different groups: irregular and regular. The regular category includes data that happens often, whereas the irregular category includes rare or unusual events that deviate from norms [5].

3 Proposed Framework

The introduced approach in this study utilizes deep learning techniques to identify anomalies in video data. This method consists of two primary components: the first focuses on feature extraction and learning, while the second is responsible for anomaly detection. Additionally, a pre-processing stage is included, which involves estimating and eliminating the background. Similar to other machine learning methods, Training and testing are the two main stages of this strategy. A subset of the dataset with only normal frames is used to learn features during the training stage. The trained model is used on a different dataset segment including aberrant frames during the testing phase.

Figure 3 depicts the overall structure of the proposed framework. As shown in the figure, there are four main types of feature learning. Some features are extracted from individual frames, while others use smaller frame sections to reduce processing time and cost.

The first feature, appearance, is related to identifying objects in each frame. A detection score is generated by comparing a frame with the ones before and after it. The final score is calculated by comparing frames and analysing average speed. The following attribute, density, is the amount of objects in a frame.

The tertiary attribute, Movement tracks object movement between frame sections, generating optical flow and forming a video sequence used for anomaly scoring. The last feature, the scene, reconstructs a scene using frame sections and a trained model. These features are combined to improve detection accuracy and produce final scores.

Data preparation

Prior to feature extraction and learning, the first stage is to anticipate and remove the backdrop. The background differs across various situations, and multiple techniques can be applied for its removal. For instance, it may consist of empty areas or roadside boundaries.

In this approach, background prediction relies on the Most Frequent Occurrence (MFO) among sections of video frames. The process starts by creating a histogram for each frame, considering pixel values and their positions. Next, the histograms of different frame sections are analysed, and the most frequently appearing values in each section are identified as the background and turned Gray. Processing speeds up and computational costs are decreased when the background is removed. This stage is crucial to the network's training process.

Feature Selection and Training Framework

The four main components of the training network include background estimation.The deep learning model designed for appearance feature extraction utilizes a stacked denoising autoencoder (SDAE) with six encoding layers and a matching six-layer decoding structure[13,14]. A 1×1 filter window, which incorporates stride and padding operations, is used to pass each video frame over the network. Binary format normalization is applied to every frame.

This SDAE model is deeper than conventional methods, containing six encoding layers and an identical six-layer decoding structure. The output of this process consists of identified objects, referred to as appearance representations. These representations are further utilized in the anomaly detection phase and also serve as input for the density estimation module, helping improve estimation accuracy.

Density estimation[15] is performed using a convolutional neural network (CNN) with an 8×8 filter window. The third component, the motion feature extraction module[13,14], identifies movement patterns by analysing the direction of objects within video patches. This deep network architecture closely resembles the appearance feature extraction model, but instead of processing full frames, it focuses on frame patches. After a patch frame is entered into the network, frames within the same patch are compared to perform optical flow computation. This stage's output, a motion representation, is essential for anomaly identification in the future.

The ultimate element is Scene Rebuilding, which is established on a restoration framework. This system design is organized as a convolutional autoencoder, integrating both a CNN-driven creator and evaluator. The creation module reconstructs the environment utilizing a 10-layer structure, recovering frames by referencing both preceding and succeeding frames within the same segment. Simultaneously, the evaluation module assesses the generated environment by comparing it to the original, thereby computing the restoration discrepancy. Significantly, the evaluator mirrors the same configuration as the creator. A high restoration discrepancy during testing signifies the existence of anomalies, whereas in the training phase, this error remains minimal, acting as a reference for anomaly identification.After training, a set of learned and combined features is established to facilitate anomaly identification.

A classifier that divides data into normal and abnormal classes is fed the learned characteristics produced by the training network during the detection step. These characteristics are presented to the network as separate and combined inputs. Since object detection and reconstruction error together serve as a trustworthy indicator for spotting anomalies, appearance-based characteristics and reconstruction error are used in tandem. The detection accuracy is improved when the reconstruction error in a particular frame is reduced.

Two additional combined features include motion representation and density maps. By acting as complementing indications, these make sure that the direction of motion and the density distribution flow are in line. This method uses straightforward deep classifiers with the SoftMax function as its classification model. During the detection phase, five classifiers with identical architectures are used, as shown in Figure 4. Each of these networks consists of five hidden layers, designed to minimize computational overhead. The final layer in each classifier is fully connected. Ultimately, each classifier determines whether a scene is normal or anomalous, assigning a score that quantifies the likelihood of anomaly presence, ranging from 0 to 1.

In addition to the classifiers, there is a reconstruction network based on auto-encoders, which follows the same architecture as the Auto-Encoder used in the training network. However, this component is pre-trained and does not include a generator module. Instead, it utilizes the previously trained generator. The final anomaly detection score and classification outcome are established by comparing the test data with the pre-trained network and calculating the difference in reconstruction errors between the discriminator and the training phase.

In the last phase, an ensemble classifier is used to make decisions that ultimately determine the detection result. This classification algorithm, which is a simple linear model, bases its judgment on the scores produced by the previous models

as well as the majority vote %. This component's architecture guarantees that a frame is labelled as abnormal if four of the six classifications classify it as such, and the final score is calculated by averaging the scores of each classifier.

Convolutional Neural Network

ConvNet, another name for a Convolutional Neural Network (CNN), is a deep learning model made specifically for processing visual data. This particular type of feed-forward artificial neural network is designed to reduce the amount of pre-processing that is required. CNNs are an improved multilayer perceptron that can learn feature representations on its own during training, doing away with the requirement for hand-crafted filters [1].

Because these networks may preserve translation invariance using a shared-weight structure, they are also known as shift-invariant or space-invariant artificial neural networks (SIANN). Motivated by biological processes [3], CNNs replicate the way the visual cortex of animals works, with neurons grouped in a way that mimics the visual processing system of the brain. Every cortical neuron is sensitive to a specific area of the visual field known as the receptive field. Complete coverage of the entire visual scene is ensured by these receptive fields' overlap.

One of the primary advantages of CNNs is their capacity to automatically extract relevant features, reducing the reliance on manual feature engineering. Unlike traditional image processing techniques that require handcrafted filters, CNNs learn optimal filters through training. This ability to autonomously discover patterns and features makes CNNs a powerful tool for image classification and computer vision applications.

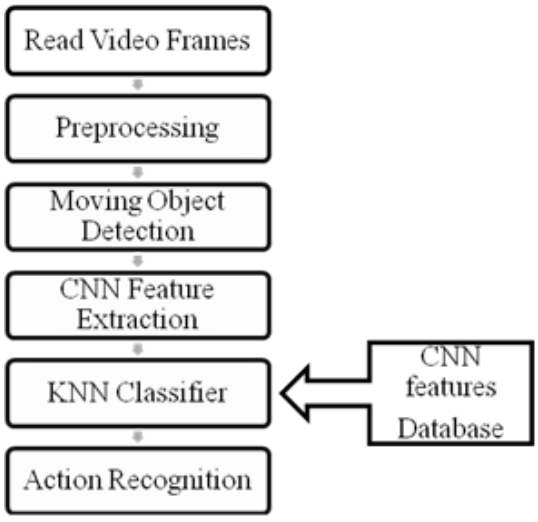

Fig 1. Flow diagram of the proposed system

They are utilized in image and video analysis, recommendation engines, and natural language understanding. A CNN is made up of multiple intermediate layers in addition to the input and output layers. Typically, these intermediate layers include of normalization, convolutional, subsampling, and tightly connected layers.

The procedure is conventionally referred to as convolution in neural networks. However, in mathematical terms, it is a cross-correlation rather than true convolution. This distinction primarily affects the arrangement of indices in the matrix, determining the specific placement of weights at each position.

Fig 2. CNN architecture

Convolutional Layer

Convolutional layers perform a filtering process on the input and forward the processed information to the next layer. This mechanism replicates how individual neurons respond to visual stimuli. Each neuron within a convolutional layer is responsible for processing data from a specific receptive field.

Although fully connected feedforward neural networks can be utilized for both feature extraction and classification, they are not ideal for image processing due to the large input size. The sheer number of neurons required, even in a shallow network, makes this approach inefficient. Since each pixel serves as an independent variable, a small image of 100 × 100 pixels would demand 10,000 weights per neuron in the subsequent layer, making the computation highly complex.

To address this issue, convolutional operations significantly reduce the number of trainable parameters, allowing deeper architectures without excessive computational load. For example, instead of treating every pixel individually, convolutional layers use shared weights within tiled regions of 5 × 5 pixels, resulting in only 25 learnable parameters. This strategy not only minimizes computational costs but also helps stabilize training by preventing the vanishing or exploding gradient issues that often arise in deep neural networks during backpropagation.

ReLU Layer

The acronym ReLU stands for Rectified Linear Units. f(x)=max(0,x) is the non-saturating activation function used in this layer. The convolutional layer's receptive fields remain unaltered, but the classification function and the network as a whole become more nonlinear.

Pooling

In convolutional neural networks (CNNs), the outputs of clusters of neurons at one layer are combined into a single neuron in the subsequent layer by means of local or global pooling layers. In the preceding layer, for instance, max pooling selects the greatest value from each cluster of neurons, while average pooling determines the mean value from each cluster. The non-saturating activation function f(x)=max(0,x) is used by the layer known as Rectified Linear Units (ReLU) to make the classification function and the network more nonlinear while preserving the receptive fields of the convolutional layer.

As a non-linear down-sampling method, pooling is an essential component of CNNs. Other non-linear functions can be used for pooling, however max pooling is the most commonly used. This approach divides the input image into non-overlapping portions, and the output is the maximum value from each sub-region. The fundamental idea is that a feature's estimated placement concerning other features is more important than its exact position.

By gradually lowering the spatial dimensions of the feature representation, the pooling layer helps to prevent overfitting by lowering the number of trainable parameters and the computational cost of the network.

In CNN architectures, pooling layers are commonly interspersed between consecutive convolutional layers. Additionally, the pooling process enhances translation invariance, further strengthening the network's ability to generalize effectively.

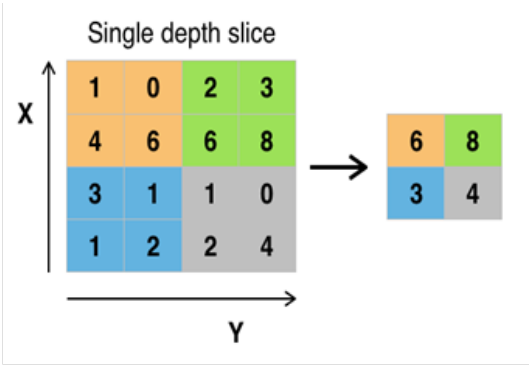

Fig 3. Maximum Pooling

Each depth slice of the input is subject to independent operation by the pooling layer, which alters its spatial dimensions without changing the depth.

A widely adopted approach involves the application of 2×2 filters with a stride of 2, effectively reducing the spatial resolution by a factor of two along both width and height, thereby retaining only 25% of the original activations. In this configuration, each max pooling operation extracts the highest value from a four-element region, ensuring that the depth dimension remains unaffected.

Alternative pooling techniques like average pooling and L2-norm pooling are used in addition to max pooling. While average pooling was historically prevalent, its adoption has declined in favor of max pooling, which has demonstrated superior empirical performance in feature extraction and preservation of salient information.

Owing to the considerable reduction in feature representation induced by pooling, contemporary deep learning architectures increasingly favor the use of smaller filter sizes or, in certain cases, the omission of pooling layers altogether to enhance feature retention and model expressiveness.

Fully connected

Fully connected layers establish a direct connection between each neuron in one layer and every neuron in the next, resembling the structure of conventional multi-layer perceptron neural networks.

In neural networks, neurons receive inputs from specific locations in the preceding layer. In a fully connected layer, each neuron is linked to all elements of the previous layer, whereas in a convolutional layer, neurons are restricted to localized regions.These areas, which are frequently square in shape (5×5), delineate the so-called receptive field. The receptive field is restricted to a smaller portion of the input in convolutional layers, while it covers the entire preceding layer in fully connected layers.

A bias term and a set of weights determine the mathematical function that is applied to each neuron's inputs to calculate its output. These weights and biases, represented as real-valued parameters, are adjusted incrementally during training to optimize the network's performance. Together, the weight vector and bias form a filter that identifies specific patterns or features within the input data.

A distinguishing characteristic of convolutional neural networks (CNNs) is the shared use of filters across multiple neurons. This shared-weight approach reduces memory consumption by ensuring that all receptive fields associated with a given filter use the same set of parameters, rather than allocating individual weights and biases to each. This approach improves computational efficiency while preserving the model's capability to learn and derive relevant features from the data.

System Development

For the simulation of system deployment, MATLAB software has been utilized. MATLAB is an advanced technical and scientific programming language that enables efficient visualization and high-speed computation. With MATLAB, data can be analyzed, processed, synthesized, and visualized effectively.

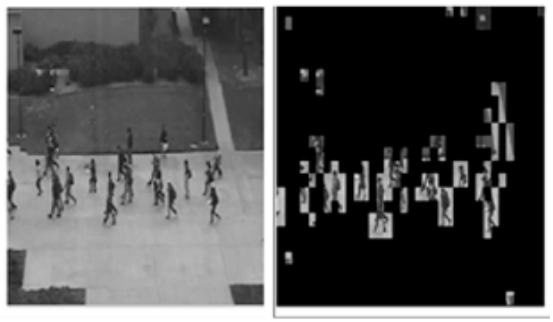

Fig 4. Density Estimation for dynamic data in video

Fig 5. Anomaly Detected in Video

Table 1. Cross-validation accuracy for UCSD Database

UCSD database	% Cross Validation Accuracy
Normal Activity	90.00 %
Anomaly Activity	90.50 %

Conclusion

This study introduces a robust deep learning framework tailored for identifying anomalies within surveillance video data. The proposed system utilizes a multi-stage architecture that includes feature extraction, model training, and anomaly classification. By employing Convolutional Neural Networks (CNNs), autoencoders, and ensemble classification strategies, the model achieves improved detection precision through the fusion of key visual indicators—such as object appearance, motion trajectories, density variations, and scene reconstruction.

Performance evaluation on the UCSD benchmark dataset indicates that the model consistently achieves high detection accuracy, with success rates exceeding 90% for both regular and irregular activities. This demonstrates the system's reliability in distinguishing abnormal behaviors from normal patterns in complex environments. Additionally, the training architecture is designed to be modular and reusable, making it adaptable for deployment across other similar surveillance contexts.

The background subtraction technique, based on identifying the most frequently occurring pixel patterns, plays a key role in reducing noise and computational demands. Combined with motion and scene reconstruction insights, this enhances the system's responsiveness and accuracy in real-time monitoring.

Overall, the proposed method highlights the effectiveness of deep learning for video surveillance applications, especially where real-time anomaly detection is critical. Potential future enhancements could involve integrating interpretability modules to explain detection outcomes, expanding the system to process real-time video feeds, and validating the model across more varied datasets for greater generalizability.

References

1) Nithesh K, Tabassum N, Geetha DD, Kumari RDA. Anomaly Detection in Surveillance Videos Using Deep Learning. In: 2022 International Conference on Knowledge Engineering and Communication Systems (ICKES). IEEE. 2023;p. 1–6. Available from: https://doi.org/10.1109/ICKECS56523.2022.10059844.
2) Chandrakala S, Deepak K. A Review of Deep Learning-Based Anomaly Detection Strategies in Videos. *IEEE Access*. 2023;11:123456–123478.
3) Wang B, Yang C. Video Anomaly Detection Based on Convolutional Recurrent AutoEncoder. *Sensors*. 2022;22(12):4647. Available from: https://dx.doi.org/10.3390/s22124647.
4) Sun F, Zhang J, Wu X, Zheng Z, Yang X. Video Anomaly Detection Based on Global–Local Convolutional Autoencoder. *Electronics*. 2024;13(22):4415. Available from: https://dx.doi.org/10.3390/electronics13224415.
5) Duong HT, Le VT, Hoang VT. Deep Learning-Based Anomaly Detection in Video Surveillance: A Survey. *Sensors*. 2023;23(11):5024. Available from: https://dx.doi.org/10.3390/s23115024.
6) Zhao M, Liu Y, Liu J, Zeng X. Exploiting Spatial-temporal Correlations for Video Anomaly Detection. *arXiv preprint* . 2022;p. 1–7. Available from: https://arxiv.org/pdf/2211.00829.
7) Wu C, Shao S, Tunc C, Satam P, Hariri S. An explainable and efficient deep learning framework for video anomaly detection. *Cluster Computing*. 2022;25(4):2715–2737. Available from: https://dx.doi.org/10.1007/s10586-021-03439-5.
8) Zeng X, Jiang Y, Ding W, Li H, Hao Y, Qiu Z. A Hierarchical Spatio-Temporal Graph Convolutional Neural Network for Anomaly Detection in Videos. *IEEE Transactions on Circuits and Systems for Video Technology*. 2023;33(1):200–212. Available from: https://dx.doi.org/10.1109/tcsvt.2021.3134410.
9) Matei A, Glavan A, Talavera E. Deep Learning for Scene Recognition from Visual Data: A Survey. In: International Conference on Hybrid Artificial Intelligence Systems;vol. 12344 of Lecture Notes in Computer Science. 2020;p. 763–773. Available from: https://doi.org/10.1007/978-3-030-61705-9_64.
10) Esmaeili F, Cassie E, Nguyen HPT, et al. Anomaly Detection for Sensor Signals Utilizing Deep Learning Autoencoder-Based Neural Networks. *Bioengineering* . 2023;10(4):405. Available from: https://doi.org/10.3390/bioengineering10040405.
11) Ma H, Zhang L. Attention-based framework for weakly supervised video anomaly detection. *The Journal of Supercomputing*. 2022;78(6):8409–8429. Available from: https://dx.doi.org/10.1007/s11227-021-04190-9.
12) Ouyang Y, Shen G, Sanchez V. Look at Adjacent Frames: Video Anomaly Detection Without Offline Training. In: Computer Vision – ECCV 2022 Workshops;vol. 13859 of Lecture Notes in Computer Science. 2023;p. 642–658. Available from: https://doi.org/10.1007/978-3-031-25072-9_43.
13) Dilek E, Dener M. Computer Vision Applications in Intelligent Transportation Systems: A Survey. *Sensors*. 2023;23(6):2938. Available from: https://dx.doi.org/10.3390/s23062938.
14) Xia X, Gao Y. Video Abnormal Event Detection Based on One-Class Neural Network. *Computational Intelligence and Neuroscience*. 2021;2021(1). Available from: https://dx.doi.org/10.1155/2021/1955116.
15) Hojjati H, Armanfard N. DASVDD: Deep Autoencoding Support Vector Data Descriptor for Anomaly Detection. *IEEE Transactions on Knowledge and Data Engineering*. 2024;36(8):3739 –3750. Available from: https://dx.doi.org/10.1109/tkde.2023.3328882.