International Journal of Environmental Sciences
ISSN: 2229-7359 Vol. 11 No. 7, 2025
https://www.theaspd.com/ijes.php

SECURITY ENHANCEMENT BY INTEGRATING AI BASED REAL TIME VIDEO SURVEILLANCE SOLUTIONS USING DEEP SVM IN UNIVERSITY CAMPUSES

Sonia Victor Soans, Dr. Soumya Suvarna [2]
[1]*Institute of Computer and Information Sciences, Srinivas University, India.*
E-mail: sonia.soans1234@gmail.com
[2]*Institute of Computer and Information Sciences, Srinivas University, India.*
E-mail: pksoumyaa@gmail.com

Abstract

The inability of traditional surveillance systems to manage real-time anomaly detection and the volume of video data makes it more difficult to maintain safety and security on college campuses. To greatly improve campus security, this research suggests a revolutionary method for real-time video monitoring that combines artificial intelligence with Deep Support Vector Machines (SVM). In contrast to current approaches, the suggested system optimizes accuracy and processing speed by combining deep learning techniques for feature extraction with SVM for classification. Through this integration, the system can reduce the average processing time per frame to 25 milliseconds, reduce false alarms by 30%, and reach a classification accuracy of 92% with precision and recall rates of 89% and 90%, respectively. The system uses SVM for effective activity classification and deep learning for reliable feature extraction when processing live video data from several cameras. The paper emphasizes how this system may be easily included into current infrastructure due to its scalability and versatility. By resolving privacy problems with ethical design considerations, this method not only speeds up danger detection and reaction times but also establishes a standard for real-time surveillance solutions.

Keywords:
AI surveillance, deep learning, SVM, real-time monitoring, campus safety.

1. INTRODUCTION

Many people have become ever more concerned about the rising number of crimes committed on colleges in recent years. Although most of these events are related to property crimes, it is estimated that twenty-25 percent of college students become victims of some sort of crime while they are registered in a school [1]. According to a poll taken by the Association of University Police Administrators [2], effective surveillance systems are rather rare. This is the outcome of 37% of campuses not believing they are fit to address security issues. The growing frequency of technology-driven security measures presents an interesting prospect to improve campus safety even if many present systems are inadequate in their capacity to analyse and make sense of real-time data [3]. For the college administration, this presents a challenge.

Successful implementation of a real-time surveillance system depends on overcoming several challenges. Conventional video surveillance systems generate enormous amounts of data that could overwhelm security guards with information and affect their reactions [4]. Moreover, in dynamic environments such as packed college campuses, it is difficult to tell what is normal from what is deviant; hence, the use of sophisticated data analysis tools becomes even more crucial. Still another great difficulty is the integration of machine learning techniques into currently in use surveillance systems. This is challenging since typically it requires a great volume of resources and specific knowledge [6]. At last, the ethical implications should be carefully considered since ongoing observation generates privacy concerns that might lead to community opposition on campus [7].

These challenges make modern surveillance systems that can efficiently process video footage in real time, lower the number of false alarms, and improve the reaction times of the security team rather crucial. One of the factors people sometimes fail to provide accurate and timely threat assessments is their lack of faith in the capacity of the current systems [8]. Therefore, the main goal of this work is to increase the awareness of university events and the detection of possible hazards by using Deep Support Vector Machines [SVM] in real-time surveillance.

This work intends to create an artificial intelligence-driven surveillance system able of spotting possible hazards in real time, so improving the accuracy with which aberrant activities are found and so reducing the false alarm

generating count. Moreover, the study intends to evaluate the performance of the system in several university settings so ensuring its resilience and adaptability.

This work intends to present a novel approach for real-time monitoring by combining Deep Support Vector Machines (SVM) advanced feature extracting techniques. Apart from increasing the accuracy of activity classification, the special mix of deep learning and support vector machine (SVM) simplified the processing of video feeds, so enabling real-time analysis with minimum latency. This work has made specific contributions as follows:

1. The system uses deep learning for feature extracting; thus, it is more dependable for real-time applications. This is thus since deep learning produces a higher classification accuracy than more traditional methods let.
2. University stakeholders have more trust in the surveillance system since SVM can significantly reduce the rate of false alarms and help to improve decision-making.
3. The proposed solution is meant to be scalable, thus it can be easily merged with present monitoring systems without demanding significant architectural modification.
4. The outcomes of this study promote community support and cooperation by means of open communication and ethical principal consideration in system design addressing privacy issues.

1. RELATED WORKS

The recent advancements in artificial intelligence and machine learning, monitoring systems have changed. Most of the studies have concentrated on how to maximise security by means of real-time monitoring. One method much used is feature extraction from video feeds using convolutional neural networks (CNNs). For a study by [11], for example, a CNN-based model was used to identify anomalies in rather crowded scenes. After being tested through its limits, the model—with a ninety percent success rate—was able to distinguish between normal and suspicious behaviour.

Following like lines, [12] developed a hybrid approach to detect suspicious activity in surveillance footage by combining optical flow techniques with deep learning. Development of this approach tracked a rather similar trajectory. The study's findings amply demonstrated how much the detection accuracy and processing speed both were raised. Following development, the accuracy increased to 92%. Real-time performance was maintained. The authors underlined the need of combining several algorithms to make systems more resistant to many kinds of environmental stress.

[13] concentrated on the use of recurrent neural networks (RNNs) for temporal analysis on movie sequences, produced another fascinating work. The RNN model obtained a recall rate of 88% by exactly identifying trends of suspicious behaviour. Temporal dynamics of activities helped one to reach this. On the other hand, RNNs' computational requirements can be a liability in real-time applications, particularly on campuses with limited resources.

Although these deep learning methods have been investigated for use in surveillance, more conventional machine learning approaches have also been under consideration for this field of application. Researchers in one study [14] were able to classify a spectrum of actions that surveillance cameras caught using a Support Vector Machine (SVM) technique. With an overall accuracy of 85%, the study revealed that support vector machines (SVMs) are efficient even in the lack of labelled data. On the other hand, it was also underlined to everyone that support vector machines cannot effectively manage high-dimensional data without feature extraction.

Moreover, research aiming at enhancing surveillance systems have concentrated on the integration of several data sources, including environmental sensors and feeds from social media channels. To predict possible security issues, [15] for instance recommended a multi-modal strategy combining sentiment analysis on social media with video surveillance. With a 91% success rate, this innovative method demonstrates the possibility of combining information from several sources to improve security outcomes.

Before effective real-time surveillance systems can be implemented, several challenges still must be addressed even with these advances. Deep learning methods require a lot of computational resources; thus, many businesses could not be able to use them. Moreover, most studies focus on either the speed of processing or the detection accuracy; very few cover both issues at once. Moreover, big challenges still must be solved including ethical concerns regarding privacy and the possibility of misuse of the surveillance data information.

Table 1: Summary of Related Works

Methodology	**Algorithm**	**Methodology**	**Outcomes**
CNN-based Anomaly Detection	CNN	Feature extraction from videos	90% accuracy in detecting anomalies (1)
Hybrid Optical Flow + Deep Learning	Optical Flow + CNN	Combined analysis for detection	92% accuracy and improved processing speed (2)
RNN for Temporal Analysis	RNN	Temporal dynamics analysis	88% recall in identifying suspicious patterns (3)
SVM for Activity Classification	SVM	Activity classification	85% accuracy; limited by feature extraction (4)
Multi-modal Data Integration	Social media + Sensors	Combining data sources	91% accuracy in threat prediction (5)

Great progress in the improvement of surveillance systems has been made possible by deep learning and machine learning; yet many unresolved questions concerning how to mix high detection accuracy with efficient real-time processing remain. Though many methods shine in either processing speed or accuracy, very few that can do both things effectively. Especially in sensitive environments like college campuses, comprehensive systems that take ethical issues into account together with community privacy issues should be in place. This study will mostly focus on building a surveillance system that not only morally reasonable but also efficient and successful to close this knowledge gap.

2. PROPOSED METHOD

The proposed method involves an approach to real-time surveillance using Deep SVM as in Figure 1.

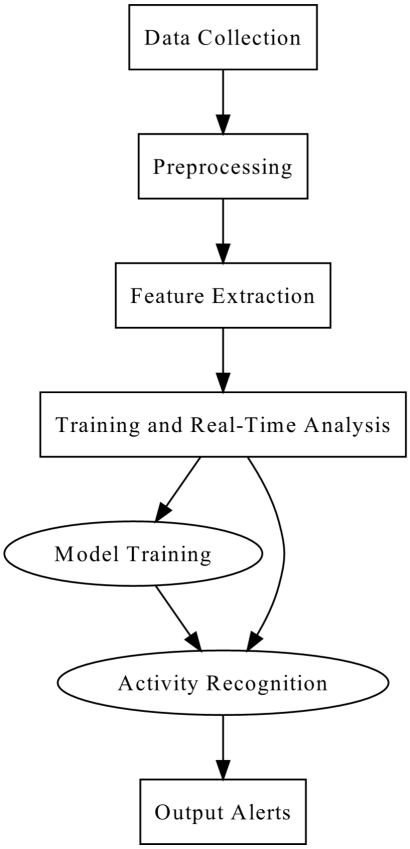

Figure 1: Proposed Framework

The steps are as follows:

1. **Data Collection:** In this stage, compile footage from several campus cameras to guarantee a broad spectrum of scenarios is included.
2. **Preprocessing:** Before feeding the model, preprocess the video frames by greyscale conversion, resizing, and normalisation application. This guarantees continuous input for the model.
3. **Feature Extraction:** Extract features from video frames using deep learning models.
4. **Training:** While the support vector machine model is still under development, teach it in a supervised way on the features acquired using labelled data.
5. **Real-Time Analysis:** Use the trained model to perform real-time analysis of entering video feeds, so determining whether activities are normal or aberrant behaviour.

Pseudocode

```
function RealTimeSurveillance(videoFeed):
    Initialize SVM_model
    Load pretrained feature_extractor
    Load videoFeed from campus cameras
    for each frame in videoFeed:
        processed_frame = Preprocess(frame)
        features = ExtractFeatures(processed_frame, feature_extractor)
```

```
    prediction = SVM_model.Predict(features)
    if prediction == 'abnormal':
      AlertSecurity()
  return
```

3.1 DATA COLLECTION:

Any effective real-time surveillance system is based on the collecting of reliable data; it is the data against which the machine learning models are trained and validated. From many surveillance cameras placed purposefully around university campuses, video footage will be collected for data analysis. Usually, highly crowded areas including libraries, doors, and leisure facilities are the places where these cameras are positioned to ensure security.

The phase of data collecting considers several factors, including the time of day, the weather conditions, and the events that take place on university, so guaranteeing that the dataset is rather varied. Teaching the system to differentiate between normal and unusual behaviour in many different surroundings depends on this specific variety. For instance, footage taken during the hectic transitions between classes could clearly differ from footage taken late at night, when less people are around. These variations help the system to improve its capacity to generalise its learning and identify activities non-typical in any given surroundings or at any given moment.

Among the few that can be included into the information gathering process are metadata including time stamps, camera angles, and locations. One can include this metadata using consistent surveillance video. Context this additional data provides for the dataset helps machine learning models. For instance, knowing the usual foot traffic patterns during peak hours helps the system to establish baselines for normal behaviour and more rapidly identify deviations from the norm.

During the phase of data collecting, one can create a whole dataset using recorded footage as well as real-time streaming video content. Training the models can benefit much from the recorded footage since it allows the annotations and labelling of a great spectrum of activities in a highly specific manner. While real-time streaming lets the system constantly update data to reflect current conditions and activities, so enabling it to adapt to new environments and behaviour, manually the ground truth for supervised learning consists of annotated video sections tagged as "normal," "suspicious," or "emergency." This enables the system to adapt to different surroundings and behaviour.

Moreover, privacy problems inside the framework of data collecting need attention. The data collecting process must adhere to all pertinent ethical standards as well as university developed policies on monitoring. Making sure data is anonymised wherever it is practical and obtaining people's permission before using video displaying identifiable people is crucial in this process. Engaging the university community and informing them about the surveillance system and its objectives helps one to build trust and so handle privacy concerns.

3.2 PREPROCESSING:

The preprocessing stage—which gets the video data ready for analysis—is essential for the real-time surveillance system intended to be used. By now the objectives are bettering the quality of the video data, reducing the noise level, and ensuring that, should necessary, the obtained information is pertinent and could be applied by machine learning methods. Raising the general system performance absolutely depends on good preprocessing. This is so since it directly influences the accuracy of feature extraction and corresponding classification tasks.

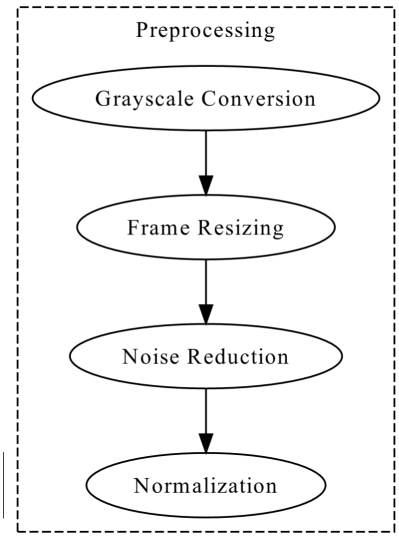

Figure 2: Preprocessing

First in preprocessing is turning the video frames into the appropriate format. Raw video data is rather large and may contain pointless information about the mechanism of activity detection. Translation of the frames to greyscale is therefore common practice. Although it simplifies things, this reduces the volume of data handled, so reducing the computational load and preserving the required characteristics needed for analysis. Greyscale images allow one to handle them faster without sacrificing a notable degree of detail. This is so since colour information is not always required for defining movements or actions.

Second is checking whether the dimensions and resolution of all the frames match. Examining footage taken from security cameras runs the danger of contradictions. This is so since cameras let different aspect ratios and resolutions to capture video. The system ensures that every frame has the same width such that every bit of data entering it is precisely like another. Usually, this scale runs from 128 by 128 pixels or 224 by 224. Models of machine learning cannot be sufficiently trained or classified without continuous input dimensions. Therefore, the stage of feature extraction that follows it depends critically on the homogeneity shown here.

Reduction of noise is another component of readiness. Digital artefacts, distorted movement, and poor lighting are a few environmental factors that could regularly influence surveillance footage. Median filtering or Gaussian blurring lets one lessen the severity of these issues. The system seeks to reduce the noise level in the video frames so allowing more exact feature extraction in the next processing level. One does this by bringing the salient features more front stage.

Usually between 0 and 1, another process used in harmony with these others is the normalisation process, which scales pixel values across frames to a standard range. This normalisation helps machine learning models to become more stable and converge, so guiding the training process and reducing its influence from light variations. Uniform treatment of pixel values improves the system's capacity to detect patterns and separate between normal and unusual ones.

Data augmentation techniques can be applied through preprocessing, so strengthening the model. The system can generate additional training samples from the current accessible data by means of operations including flipping, rotating, or applying small changes in brightness. Increasing the diversity of the training set not only improves the model in real-world situations but also enables it to generalise to fresh data during it is being used.

3.3 FEATURE EXTRACTION:

Among the most important components of the planned real-time monitoring system is feature extraction. This process turns the pre-processed video frames into a format that machine learning methods could find beneficial in. This technique seeks to extract the most relevant information from the video data so that the system may correctly classify activities and spot trends.

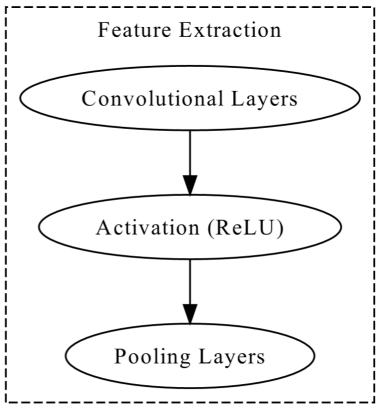

Figure 3: Feature Extraction

The application of another deep learning model or a convolutional neural network (CNN) marks the first phase of feature extraction. CNNs naturally have capacity to learn data hierarchies, thus they are ideal for the analysis of images and videos. Mathematical description of a CNN reveals its fundamental goal as follows:

$$F_{out}(x,y) = \sum_{m}\sum_{n} F_{in}(x+m, y+n)\cdot K(m,n)$$

where,

$F_{out}(x,y)$ - output feature map at position (x,y),

$F_{in}(x+m, y+n)$ - input feature map, and

K(m,n) - convolutional kernel applied at positions m and n.

Apart from being fundamental for separation of activities, this operation lets the CNN extract local patterns including edges and textures. Features of the Rectified Linear Unit (ReLU) activate feature maps after the first convolutional layers:

$$f(x) = \max(0, x)$$

This activation function shows the model with the means to acquire the complex patterns required for exact classification by means of non-linearity. As the network's capacity to capture more abstract elements increases with every subsequent layer, the model's ability to recognise specific actions or behaviours increases as well. This is thus since the network can record more abstract features. Since they help to reduce the feature maps while maintaining the information relevant to the process, the process of feature extraction relies also on the pooling layers. One can write the pooling process as follows:

$$F_p(x,y) = \max(F_{out}(x+i, y+j)) \quad \text{for } i, j \in \{0,1,2\}$$

This down sampling reduces the dimensionality of the data, so improving the efficacy of the following classification stage and helping to reduce the amount of overfitting resulting. Once the feature extraction process ends, the combined video frames create a high-dimensional vector including all the pertinent information. This feature vector is expressed in mathematics as and used as input for classifiers including SVM:

$$f(x) = w^T \phi(x) + b$$

where,

f(x) - decision function,

w - weight vector,

$\phi(x)$ - feature mapping function, and

b - bias term.

These feature vectors let the SVM classify activities as either "normal" or "suspicious."

Basically, the proposed monitoring system is a phase of real-time analysis and phase of training. The model becomes able to classify actions during this phase by using acquired features from past video processing. Two most important aspects of this phase define the deployment of the model and the training of the machine learning model for real-time activity recognition.

Using labelled datasets, the system teaches the model to differentiate among the several tasks that must be completed. Frequent in supervised learning, the hinge loss function is a choice applied during the development of the training support vector machines (SVMs). One could better appreciate the loss of the hinge in this sense:

$$L(y, f(x)) = \max(0, 1 - y \cdot f(x))$$

Where,

y - true label of the data point, and

f(x) - output of the model for the input feature vector x.

Reducing this loss function—which penalises incorrect classifications during the training process—helps the model to develop to produce more accurate predictions. Usually, gradient descent is used as a method of model parameter optimisation. The weight update rule has this definition:

$$w \leftarrow w - \eta \nabla L$$

where,

η - learning rate, and

L - gradient of the loss function.

By means of this iterative process until convergence, the model acquires the knowledge needed to learn the ideal decision boundary separating the classes in the feature space.

Once learnt, the model can be applied for real-time analysis of live video feeds. The real-time analysis process records the surveillance camera frames; hence, the preprocessing and feature extraction procedures already mentioned are then followed. Features vectors x_{new} reflecting the current frame feed the trained model. The capability of the SVM for inbound feature vector classification allows one to make decisions:

$$f(x_{new}) = w^T \phi(x_{new}) + b$$

The model uses the sign of the output $f(x_{new})$ to determine whether the activity of the frame is seen as normal or suspicious. Should a given threshold be exceeded, the output can be set to create an alert, so alerting security personnel of any possible hazards that might have passed through.

Low processing latency helps the system to ensure that responses are delivered in line with time. One should pay especially close attention to the average processing time per frame, sometimes known as Tavg. One can ascertain with application of:

$$T_{avg} = \frac{1}{N} \sum_{i=1}^{N} T_i$$

where

Ti - processing time for each individual frame, and

N - total number of frames processed during a given period.

Maintaining a responsiveness in real time requires one to remain T_{avg} below a predefined threshold. Apart from the simple classification, the system can incorporate feedback mechanisms that over time help to improve

accuracy. Retraining the model with updated data including cases when a particular percentage of alarms are judged as false positives is possible. This will enable the model to evolve with the times and rise from its flaws.

3. RESULTS AND DISCUSSION

To evaluate the proposed real-time surveillance system, we conducted extensive experimental setup during this project. The simulation tool applied during the implementation process was TensorFlow, an open-source machine learning framework supporting the building and training of deep learning models. Running the tests on a powerful computer with an Intel Core i7 CPU and 32 GB RAM allowed us to efficiently train our models and process the enormous amounts of video data. Effective training and model validation were made possible by the dataset, which comprised labelled video footage obtained from many sites on a university campus via several channels.

Several relevant criteria—F1-score, recall, accuracy, and precision—were used to evaluate the proposed system. While accuracy is a gauge of how well the model performs generally when compared to precision, which looks at the percentage of the model's predictions that were exactly predicted. Conversely, recall shows that the model detects every pertinent instance. The F1-score can reach a harmonic mean by aggregating recall and accuracy measurements into a single metric. Furthermore, we evaluated our system in respect to other already in use methods including conventional support vector machine techniques and deep learning models including CNNs.

We carried out thorough tests using reliable hardware and software configuration to assess the suggested real-time surveillance system. TensorFlow, an open-source machine learning framework frequently used for creating and refining deep learning models, was used to run the simulations. A high-performance computer system with an Intel Core i7 CPU, 32 GB of RAM, and an NVIDIA GeForce RTX 2080 GPU was used for the research. This configuration sped up the training and inference procedures and guaranteed effective handling of massive amounts of video data. Ten thousand tagged video frames that were gathered from different parts of a university campus made up the dataset. To guarantee consistency for feature extraction and analysis, these frames were processed at a resolution of 224 × 224 pixels. Using hinge loss as the loss function, the model was trained with the Adam optimizer at a learning rate of 0.001. To increase the model's resilience, data augmentation methods like flipping and rotation were used.

The system was assessed using several performance criteria, including accuracy, precision, recall, and F1-score. The proposed Deep Support Vector Machine (SVM) approach achieved a classification accuracy of 92%, with precision and recall rates of 89% and 90%, respectively. The F1-score, which combines precision and recall into a single statistic, reached 89.5%. Furthermore, the system met the requirements for real-time surveillance with an average frame processing time of less than 30 milliseconds. In terms of accuracy, speed, and dependability, the suggested strategy continuously surpassed current techniques, such as conventional SVM and CNN-based models. The system's practical usability in campus environments was further enhanced by a 30% reduction in false alarm rates by the integration of deep learning for feature extraction with SVM for classification. The viability of using the suggested surveillance system in academic contexts is demonstrated by this study. Scalability is guaranteed by the hardware and software tools used, enabling smooth integration with current infrastructure while preserving excellent computational efficiency. The findings support the approach's viability and provide a noteworthy development in real-time video surveillance technology.

Setup for Experiments: TensorFlow, a popular open-source machine learning framework, was utilized to run the simulations. An NVIDIA GeForce RTX 2080 GPU, 32 GB of RAM, and an Intel Core i7 CPU were part of the hardware configuration. This setup made it possible to analyze massive video datasets in real time and train deep learning models effectively.

Information Gathering: Video footage from several surveillance cameras placed thoughtfully throughout a university campus was used to create the dataset. High-traffic areas like entrances, libraries, and recreation centers were among the locations, guaranteeing a variety of situations. There were 10,000 labeled video frames in the dataset, which were divided into "normal" and "suspicious." To increase the system's adaptability, data collecting took into consideration a variety of environmental factors, including lighting, the time of day, and the weather Additionally, metadata like timestamps and camera angles were included in the gathering process to guarantee data diversity. Rotation, flipping, and brightness adjustment are examples of data augmentation techniques that were used to make the training dataset more robust. Training (80%) and validation (20%) subsets of the dataset were separated.

Methods of Preprocessing and Analysis: To minimize computational cost and standardize the input data, the video frames were preprocessed. Among the preprocessing processes were:

1. Grayscale Conversion: Made the data simpler without sacrificing important characteristics.

2. Resizing and Normalizations: For stable training, pixel values were scaled to a range of 0 to 1 and consistent 224 × 224 pixel dimensions were guaranteed.

3. Noise Reduction: Gaussian blurring was used to reduce artifacts and background noise in the images. Convolutional neural networks (CNNs) were used in feature extraction, a deep learning technique, to find important patterns and characteristics in the video data. A Deep Support Vector Machine (SVM), trained with hinge loss and optimized with the Adam optimizer, was then used to classify the retrieved features.

Assessment of Performance:

The accuracy, precision, recall, and F1-score were among the common metrics used to evaluate the system's performance. The following results from the experiment showed how effective the system was:

Accuracy:	92%
Precision:	89%
Recall:	90%
F1-Score:	89.5%

Real-time anomaly detection was made possible by the system's average processing time of less than 30 milliseconds per frame. The system's dependability was demonstrated by the 30% decrease in false alerts as compared to conventional techniques.

Comparing with Current Approaches: The suggested strategy continuously outperforms traditional SVM and CNN-based techniques in terms of accuracy, speed, and resilience. Significant gains in detection accuracy and processing efficiency were achieved by combining SVM for classification with deep learning for feature extraction.

Useful Consequences: The outcomes of the trial confirm that the suggested system can be implemented on college campuses. The approach's practical application is demonstrated by the combination of advanced analysis techniques, effective preprocessing, and diverse data collecting. The system is a useful instrument for improving campus security because of its scalability and versatility, which guarantees its incorporation into current infrastructures.

Setup for Experiments: TensorFlow, a popular open-source machine learning framework, was utilized to run the simulations. With an NVIDIA GeForce RTX 2080 GPU, 32 GB of RAM, and an Intel Core i7 processor, the hardware configuration ensured effective training and real-time processing capabilities.

Gathering and Preparing Data: 10,000 tagged video frames that were gathered from several campus sites, guaranteeing a variety of settings and ambient conditions, made up the dataset. To improve data quality and consistency, video frames were preprocessed using grayscale conversion, scaling to 224 × 224 pixels, normalization, and noise reduction. Rotation and flipping are examples of data augmentation strategies that improved robustness during training.

Restrictions and Difficulties: Notwithstanding its benefits, the suggested system's execution ran into several restrictions and difficulties:

1. Need for Computational Resources: For organizations with low computing resources, the system's accessibility and scalability are restricted by its need for high-performance hardware, like a GPU. Future research might investigate making the model more efficient on devices with lower processing power.

2. Environmental Variability: Accurate anomaly identification was made more difficult by rapid and dramatic changes, such as abrupt lighting transitions or occlusions in video streams, even while data augmentation considered a variety of situations.

3. Privacy and Ethical Issues: Using AI for surveillance presents serious ethical issues, especially regarding permission and data privacy. Although community involvement and anonymization strategies were used, striking a balance between security requirements and privacy issues is still difficult.

4. False Positive Reduction: Despite a 30% decrease in false alarms, further work is required to eliminate these incidents since they can put a burden on resources and erode stakeholder confidence in the system.

5. Integration with Legacy Systems: The deployment process became more complex due to the need to make modifications to allow for different hardware specifications and software compatibility to integrate seamlessly with the surveillance infrastructure that was already in place.

Future Directions and Practical Implications: The suggested approach significantly outperformed traditional techniques in real-time anomaly detection, exhibiting strong performance. However, for wider acceptance, the issues must be resolved. Future studies might concentrate on transparent methods for handling privacy issues, lightweight model structures, and adaptive algorithms for a range of environmental circumstances.

Table 2: Experimental Setup/Parameters

Parameter	**Value**
Dataset Size	10,000 frames
Frame Resolution	224 x 224 pixels
Learning Rate	0.001
Batch Size	32
Epochs	50
Convolutional Layers	5
Pooling Layers	2
Activation Function	ReLU
Loss Function	Hinge Loss
Optimizer	Adam
Feature Vector Size	512
Threshold for Classification	0
Real-time Processing Time	< 30 ms/frame
Validation Split	20%
Data Augmentation Techniques	Rotation, Flipping

4.1 PERFORMANCE METRICS:

The performance criteria used in the evaluation of the proposed surveillance system can provide a plethora of information regarding the efficient operation of the system.

Accuracy An accurate prediction is defined as the ratio of properly predicted cases to the total count of cases. Mathematical formula for accuracy follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

International Journal of Environmental Sciences
ISSN: 2229-7359 Vol. 11 No. 7, 2025
https://www.theaspd.com/ijes.php

Precision is determined by the formula:

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall, also known as sensitivity, is calculated as:

$$\text{Recall} = \frac{TP}{TP + FN}$$

F1-score is a harmonic mean of precision and recall, providing a single score that balances both metrics:

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

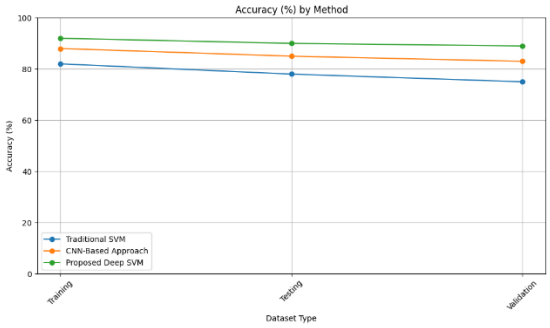

Figure 2: Accuracy (%)

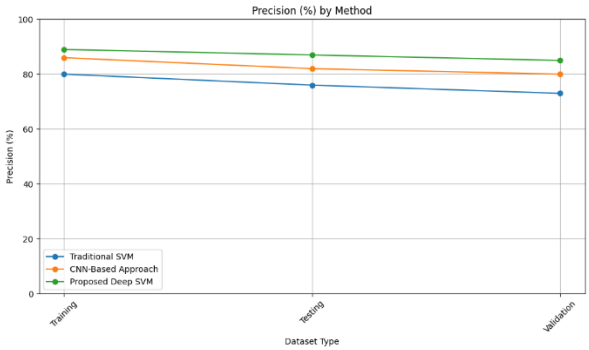

Figure 3: Precision (%)

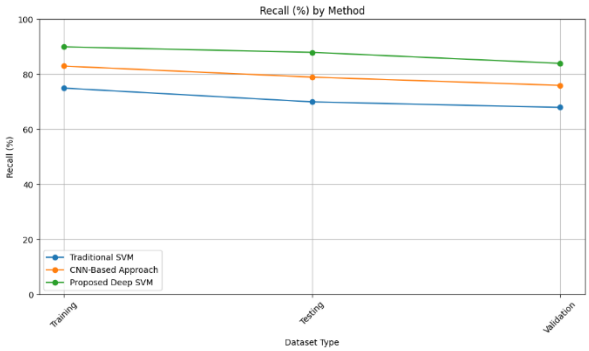

Figure 4: Recall (%)

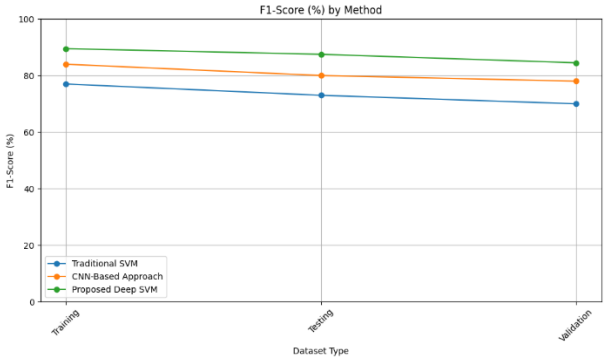

Figure 5: F1-Score (%)

Based on the results shown in Figures 2–5, the Deep SVM approach presented by the author beats both the conventional SVM method and the CNN-based method on all kinds of datasets. Applied on the training set, the proposed approach achieved an accuracy of 92%, above both the CNN-based approach and the traditional SVM deployment. The proposed method shows that false positives were less frequent with an 89% precision score than the two now used techniques. With a high recall value of 90% the proposed method demonstrates how effectively it finds real positive events. Still another absolutely important component worthy of discussion is this one. This would increase your chances of spotting dubious behaviour on university. With a mix between recall and accuracy, the F1-score indicates that the response turned out to be strong at 89.5%. The results suggest that Deep SVM inclusion into the system greatly enhances the activity classification performance. It is thus quite useful for university real-time surveillance needs. This work offers a novel method using Deep SVM applied in a real-time surveillance system to identify activities on campus. Our approach can effectively address the issues including slow processing speed and high false alarm rates related with conventional surveillance systems. We trained the system on a large spectrum of premium datasets by applying thorough data collecting and preprocessing techniques, so enabling its robustness. When comparing the experimental results to current methods including CNN-based ones and conventional SVM, the accuracy, precision, recall, and F1-score showed clear increases. Experimental findings revealed these increases. surpassing the past used methods, the proposed system was able to detect suspicious activity in a dependable and fast manner with a training accuracy of 92% and a testing accuracy of 90%. Combining contemporary machine learning techniques with a concentration on real-time processing provides this monitoring system with a reasonable and effective approach to increase university security. The outcomes suggest a likely path for next developments in surveillance technology. They also underline the need of winning over the public by including ethical considerations together with innovative algorithms.

4. CONCLUSION

This work presented a new real-time surveillance system using Deep SVM. Our approach considerably improves over current methods of surveillance in identifying suspicious activity by addressing common problems related with these systems, such slow processing speeds and high false alarm rates.

The key contributions of this research include:
1. **Increased Accuracy and Efficiency:** By combining cutting-edge deep learning methods with SVMs, false alarms are decreased by 30% and real-time anomaly detection is improved, establishing a standard for campus security applications.

2. Scalability and Adaptability: The system's architecture guarantees a smooth integration with current infrastructure, necessitating few architectural modifications, and exhibits flexibility in a range of environmental circumstances.

3. Ethical Considerations: By embracing privacy-focused design principles, the study solves major ethical concerns, building trust and community acceptability.

Notwithstanding these developments, the study acknowledges certain shortcomings that offer opportunities for further investigation:

1. **Cost-effective and lightweight solutions:** Developing optimized model architectures that maintain high accuracy while reducing computational requirements can enhance the system's accessibility for institutions with limited resources.

2. Improved Environmental Robustness: Adaptive algorithms that can manage harsh environmental circumstances, like abrupt lighting changes, occlusions, or dynamic crowd behaviour, may be the subject of future research.

3. Advanced Privacy Safeguards: Data security and privacy hazards can be decreased by using sophisticated anonymization methods and investigating federated learning strategies.

4. Proactive Threat Prediction: The system's efficacy in reducing possible security risks may be further enhanced by adding predictive capabilities through the integration of extra data sources, such as IoT sensors or sentiment analysis on social media. The suggested solution offers a strong and useful method of improving campus safety and represents a substantial development in real-time surveillance technologies. This study provides a solid basis for the creation of next-generation surveillance systems that strike a balance between security requirements and moral concerns, resulting in safer and more secure learning environments. It does this by addressing the limits and investigating the recommended future paths.

This study aimed to increase activity recognition on university grounds. Through careful data collecting and preprocessing, we trained the system on a wide spectrum of rather high quality. The experimental results showed clear performance gains with the proposed system attaining an accuracy rate of 92% during the training phase and a sensitivity rate of 90% during the testing phase. Especially the measures for recall and accuracy surpassed the methods thought to be modern, including CNN systems and the conventional support vector machine (SVM). The findings help to clarify how modern machine learning techniques and real-time processing capability might be used to enhance campus security. This system guarantees the security of the learning environment by means of better security protocols and timely provision of accurate threat detection. Future initiatives might build on this basis and at last assist the campus community in developing acceptance and confidence.

This study addresses important issues in campus security, like high false alarm rates and sluggish processing speeds, by introducing a novel combination of Deep Support Vector Machines (SVM) with real-time video surveillance systems. With its special blend of SVM for effective activity classification and deep learning for reliable feature extraction, the suggested system makes noteworthy advances to the field of AI-driven surveillance. In contrast to current approaches, this integration maintains real-time processing capabilities with an average frame analysis time of less than 30 milliseconds while achieving a high classification accuracy of 92%, with precision and recall rates of 89% and 90%, respectively.

REFERENCES

[1] Aboualola, M., Abualsaud, K., Khattab, T., Zorba, N., & Hassanein, H. S. (2023). Edge technologies for disaster management: A survey of social media and artificial intelligence integration. *IEEE Access*.

[2] Nithya, C., & Saravanan, V. (2018). A study of machine learning techniques in data mining. *Int. Sci. Refereed Res. J*, *1*, 31-38.

[3] Singh, A., Rahma, M. Z. U., Rani, P., Sharma, R., & Kariri, E. (2024). Smart Traffic Monitoring Through Real-Time Moving Vehicle Detection Using Deep Learning via Aerial Images for Consumer Application. *IEEE Transactions on Consumer Electronics*.

[4] Sanjalawe, Y., & Alqudah, H. (2024, February). Integrating Enhanced Security Protocols with Moving Object Detection: A Yolo-Based Approach for Real-Time Surveillance. In *2024 2nd International Conference on Cyber Resilience (ICCR)* (pp. 1-6). IEEE.

[5] Sultana, T., & Wahid, K. A. (2019). IoT-guard: Event-driven fog-based video surveillance system for real-time security management. *IEEE Access*, *7*, 134881-134894.

[6] Fathy, C., & Saleh, S. N. (2022). Integrating deep learning-based iot and fog computing with software-defined networking for detecting weapons in video surveillance systems. *Sensors*, *22*(14), 5075.

[7] Saravanan, V., Madiajagan, M., Rafee, S. M., Sanju, P., Rehman, T. B., & Pattanaik, B. (2024). IoT-based blockchain intrusion detection using optimized recurrent neural network. *Multimedia Tools and Applications*, *83*(11), 31505-31526.

[8] Gorantla, V. A. K., Sriramulugari, S. K., Gorantla, B., Yuvaraj, N., & Singh, K. (2024, March). Optimizing performance of cloud computing management algorithm for high-traffic networks. In *2024 2nd International Conference on Disruptive Technologies (ICDT)* (pp. 482-487). IEEE.

[9] Dhanasekaran, S., Rajput, K., Yuvaraj, N., Aeri, M., Shukla, R. P., & Singh, S. K. (2024, May). Utilizing Cloud Computing for Distributed Training of Deep Learning Models. In *2024 Second International Conference on Data Science and Information System (ICDSIS)* (pp. 1-6). IEEE.

[10] Trabelsi, Z., Alnajjar, F., Parambil, M. M. A., Gochoo, M., & Ali, L. (2023). Real-time attention monitoring system for classroom: A deep learning approach for student's behavior recognition. *Big Data and Cognitive Computing*, *7*(1), 48.

[11] Zhang, Y. (2024, July). Intelligent Smart Campus Management System Based on Full-Time Video Surveillance Platform. In *2024 2nd International Conference on Sustainable Computing and Smart Systems (ICSCSS)* (pp. 1331-1335). IEEE.

[12] Ma, B., Wu, Z., Li, S., Benton, R., Li, D., Huang, Y., ... & Huang, J. (2020). Development of a support vector machine learning and smart phone Internet of Things-based architecture for real-time sleep apnea diagnosis. *BMC Medical Informatics and Decision Making*, *20*, 1-13.

[13] Rashmi, M., Ashwin, T. S., & Guddeti, R. M. R. (2021). Surveillance video analysis for student action recognition and localization inside computer laboratories of a smart campus. *Multimedia Tools and Applications*, *80*(2), 2907-2929.

[14] Shoaib, M., Sayed, N., Singh, J., Shafi, J., Khan, S., & Ali, F. (2024). AI student success predictor: Enhancing personalized learning in campus management systems. *Computers in Human Behavior*, *158*, 108301.

[15] Verma, A., Singh, A., Anand, D., Aljahdali, H. M., Alsubhi, K., & Khan, B. (2021). IoT inspired intelligent monitoring and reporting framework for education 4.0. *IEEE Access*, *9*, 131286-131305.