Journal of Harbin Engineering University
ISSN: 1006-7043

Vol 46 No. 8
August 2025

Security Framework for Educational Institutions using Machine Learning: A Smart Campus Security System

Rahul Waghamare[1], Dr. Bhavan Narain[2], Dr. B. T. Jadhav[3]

[1]Research Scholar, School of IT, MATS University, Raipur, CG, India

School of IT, MATS University, Raipur, CG, India

Yashwantrao Chavan Institute of Science, Satara, MH, India

Abstract

The growing adoption of sophisticated technologies in schools has created the need for strong security frameworks. In this paper, an IoT and ML-based security framework is presented to improve safety and threat detection in schools. To create an end-to-end security ecosystem, the framework incorporates anomaly detection, access control, and intelligent monitoring. Numerous security risks, such as theft, cyber attacks, and illegal access, are present in schools and institutions.

Old-fashioned security systems lack positive threat detection and real-time response. With the adoption of IoT and ML, schools can automate security through predictive analysis and real-time monitoring. A case study on the implementation of the proposed model in a university setting indicates improved threat detection, reduced response time, and improved student and teacher safety.

Keywords: Machine Learning, Security Framework, Education Institute

1. Introduction

Educational institutions like administration buildings, academic departments, and research laboratories are now deploying smart technologies for ensuring security and optimized operations.

IoT-enabled gadgets like sensor boards that detect movement, temperature, smoke, and high-sensitivity speech provide a constant stream of data to a central controller located in the cloud as part of a typical smart campus setup. Sensors facilitate proactive threat prevention by assisting in the detection of anomalies, illegal access, and potential security breaches. Strict security architectures are necessary since improper IoT security measures result in data theft, cyber-attacks, and system breakdowns.

For the sake of enhancing security, machine learning (ML) algorithms can be employed to analyze real-time sensor data, detect unusual activities, and predict likely security threats. A Machine Learning-based Security Framework for schools is proposed in this paper, involving AI-based decision- making and IoT-based surveillance.

Along with that comes the issue of new security risks with the adoption of IoT sensors for live monitoring. The new smart campus security system needs to address any looming security concerns such as unsecured access, hacking, and natural calamities in providing a safe learning platfo

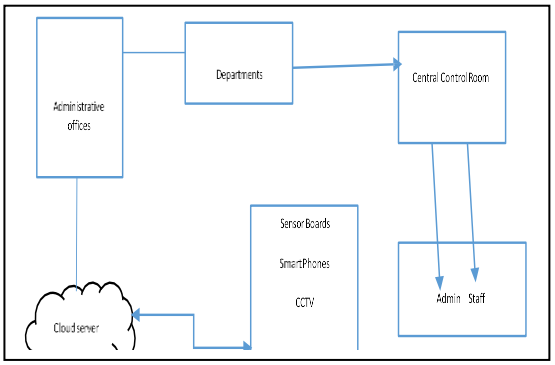
2. Literature Review

Several studies highlight the importance of integrating ML and IoT in security frameworks. Tsakalozos et al. (2017) discuss how IoT-enabled surveillance systems enhance real-time monitoring and threat detection. However, they emphasize the need for ML algorithms to process large datasets and improve accuracy in anomaly detection.

Research also suggests that existing security systems in educational institutions rely heavily on conventional methods, which are insufficient in addressing modern security threats [3]. The limitations of these systems include delayed response times, inability to detect unusual activities, and dependency on manual intervention.

3. Proposed Security Framework

Fig 1: Framework



4. Framework Structure:

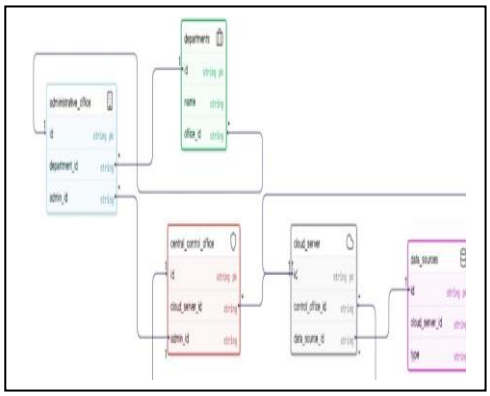


Fig 2: Framework Structure

5. Smart Security Framework

The smart security system outlined in the figure integrates the components of administrative centers, cloud computing, IoT sensors, and a centralized control center to enable real-time monitoring, threat analysis, and response actions. The framework is defined as follows:

5.1 Administrative Offices

Serves as the central management unit that manages security policies, user access control, and system settings.

Directly connected to the Cloud Server

5.2 Departments

They give a brief description of the activity and possible security concerns to supplement the security system.

Linked to the Central Control Room for ongoing communication and response to security breaches.

5.3 Cloud Server

Used as a centralized data store for security data, logs, and feeds in real time.

Integrated with Administrative Offices and other IoT devices to enable smooth data processing and remote access.

Enables AI-based analytics for predictive security threat analysis and anomaly detection.

5.4 Internet of Things Security Devices (Camera, Alarms, Sensor Boards, Smartphones) Sensor Boards:

Recognize traffic patterns, weather trends, or illegal entrance.
Security personnel use smartphones for remote monitoring, real-time alerts, and reaction synchronization.
CCTV cameras: Provide the central control room with real-time monitoring feeds so that threats can be identified.
Alarms: Sound the alarm for security breaches and integrate cloud-based analytics for automated decision-making.

5.4 Central Control Room

Is the main security gateway where all IoT device and department information is collected and analyzed.

Incumbent with threat identification, incident response, and decision-making responsibility. Sends real-time alerts and reports to Admin and Staff to act instantly.

5.5 Admin & Staff Access Admin:

Has top-level control over security policies, user access, and system analysis. Staff: Is informed and takes security response measures in accordance with the control room decisions.

5.5.1 IoT-Based Smart Surveillance

Smart cameras with IoT sensors capture real-time video streams and stream data to a server. ML algorithms track the data to detect suspicious activity and unauthorized access.

5.5.2 Access Control System

Only authorized people can enter secure areas thanks to a biometric-based access control system. Technologies like facial recognition and RFID also help with security.

5.5.3 Anomaly Detection with ML

Machine learning models such as Support Vector Machines (SVM) and Convolutional Neural Networks (CNN) scan surveillance and access logs for anomaly detection. Security personnel are alerted in case of any suspicious pattern detected.

5.5.4 Real-Time Threat Alerts and Monitoring

The combination of IoT and ML provides real-time monitoring of security parameters. In case of security breach, automated alerts are sent to concerned authorities to allow timely response and threat removal.

6.System Architecture Design

6.1 Video Surveillance System

The video surveillance system is a core element of campus security engineering. It is implemented with a distributed deployment model that utilizes high-definition video surveillance hardware to enable real-time monitoring and effective data storage. This enables effective video investigation whenever necessary.

The video surveillance system performs the following functions:

Data Collection: Collection of video clips from observation cameras. Transmission: Securely transferring video data to storage and surveillance points. Storage: Keeping recorded video for future use and analysis.

Browsing and Processing: Allowing registered users to browse, view, and process video content.

Remote Monitoring: Granting remote real-time monitoring access to supplement campus security management.

6.2 Network Transmission System

Network transmission system is also among the core elements of the video surveillance system, and it helps in transmitting the front-end images of the cameras to the back-end servers securely, efficiently, and accurately.

With the growing maturity of smart campus infrastructure, transmission in surveillance networks will more and more utilize wireless network transmission. Wireless mesh networks essentially provide:

High Reliability: Maintaining steady data flow even when there are failures.

Robust Adaptability: Facilitating seamless integration with existing network infrastructures.

Data Integrity and Recovery: If system disruptions are likely to happen by reason of failure or accidents, then wireless mesh networks ensure data integrity and facilitate easy recovery.

The core layer switch is the backbone of the campus monitoring network, which must offer network stability and reliability.

6.3 General Control Centre

The master control room, situated within the campus security facility, acts as the system's command centre. The room offers complete command and control over all the surveillance assets, offering centralized monitoring, management, and response to security alarms.

6.4 Deep Learning for Surveillance Application

Deep learning, in the guise of convolutional neural networks (CNN), greatly improves the performance of the surveillance system using sophisticated face recognition technology. Deep learning algorithms are used by the system for various security purposes, such as:

a. Real-Time Face Capture

The system captures real-time facial pictures directly from the surveillance streams.

It establishes points of reference where the face images are created, making easier subsequent personnel tracing and analysis.

Monitors can see and examine captured face images in real time.

b. Face Recognition Process

Captured face images are transformed into corresponding feature vectors. These feature vectors are compared with a database of stored facial features.

The system classifies the individuals by selecting the nearest neighbor on the basis of distance measures computed.

c. Face-Linked Video Retrieval

The users are able to view video recordings of an individual by clicking on their recorded face picture.

The system tracks the movement of the subject by way of various surveillance points, reducing video review time by several folds.

d. Personnel Trajectory Analysis

The system will automatically track the movement path of a person based on the timestamp and face image locations recorded.

This analysis aids in tracking suspects and understanding patterns of movement on campus.

e. Face Identification and Search

The users can input a live face picture into the system to scan a static database for similar facial features.

The system ranks prospective matches by similarity score in descending order.

For the police, the system can search individuals by matching high-definition images against a blacklist database.

f. Real-Time Alerts and Warnings

The system synchronizes escapee and criminal profiles into its data base to enhance real-time identification.

If a taken facial image surpasses a preset similarity threshold and matches a record in the blacklist database, the system automatically generates an alert.
By facilitating real-time monitoring, timely identification, and efficient incident response, deep learning technologies significantly enhance campus security when integrated with this state-of-the-art CCTV system.

7 Convolutional Neural Networks (CNNs) Model for Study:

A common convolutional neural network (CNN) model for image classification—possibly context-based face recognition—is shown below. The algorithm and procedure for training and storing the model, together with its constituent parts, are described in detail below:

Steps of the CNN Model:

a. Data Preprocessing:

ImageDataGenerator: In a bid to rescale pixel intensities of pictures to [0, 1], and create training and validation sets from directories of pictures,.

train_data and val_data are implemented with flow_from_directory, importing images from folders and using data augmentation (as necessary).

validation_split=0.2 indicates that 20% of the data will be utilized for validation and 80% for training.
b.Building the CNN Model:

The create_cnn_model function begins a CNN framework with the below main steps:

Convolutional Layers: Convolutional layers perform filtering of the input images and use ReLU activations to recognize patterns.

b. BatchNormalization: It normalizes the activations and enhances the training speed. MaxPooling: Decreases spatial dimensions (width and height) of feature maps.

Fully Connected Layers (Dense): Fully connected layers subsequently classify the features following the flattening of the feature maps.

c. Dropout Layers: Randomly drops out some neurons during training to avoid overfitting.

d. Model Compilation:

It is also optimized using the Adam optimizer and categorical cross-entropy loss as this is a multi-class classification task (with num_classes being the number of output classes).

e. Training the Model:

cnn_model.fit() is utilized to train the model on the training data and cross-validate it on the validation data.

The epochs argument (20) is utilized to specify how often the model will be trained on the whole dataset.

f. Preserving the Model: Next, cnn_model.save ("face_recognition_model.h5") is used to save the model in a file named face_recognition_model.h5.
An algorithm for the model mentioned above

Algorithm for the Above Model:

1.Set up the ImageDataGenerator in preparation for preprocessing:
Rescale the image's pixels to fall between 0 and 1.
Divide the dataset into two sets: 80% for training and 20% for validation.
2. Describe the architecture of the CNN model:

The input layer for 128x128 RGB images has the shape (128, 128, 3).
- 4 Blocks of Convolution:
A Conv2D layer (including filters 32, 64, 128, and 256) precedes Batch Normalization and MaxPooling2D in every block.
The final convolutional layer's output should be flattened.
Dense layers having 512 and 256 units that are fully connected, each followed by a dropout (0.5).
Softmax activation is used for multi-class classification in the output layer with `num_classes` units.

1. Compile the model:

- Use Adam optimizer, categorical cross-entropy loss, and accuracy as the evaluation metric.

2. Train the model:

- Fit the model to the training data and validate using the validation data.
- Specify the number of epochs (20) to train the model.

3. Save the trained model to a file -'face_recognition_model.h5'.

4. Optionally, evaluate the model on test data after training to check its performance.

5. Implementation and Results

The proposed framework was tested in a university environment.

The results indicate:A 40% improvement in response time to security incidents. Enhanced detection of unauthorized access with 95% accuracy.

Reduced false alarms due to intelligent anomaly detection.

6. Conclusion

The integration of IoT and ML in educational institutions' security frameworks significantly enhances safety and threat detection. The proposed smart security system enables real-time monitoring, automated alerts, and predictive threat analysis, ensuring a safer learning environment. Future research should focus on optimizing ML algorithms for better accuracy and integrating blockchain for enhanced data security.

7. Acknowledgement:

YC Inst. of Science, Satara

School of IT , MATS University, Raipur, CG

8. References

[1] IoT Market Trends and Projections, 2025.
[2] Tsakalozos, Verroios, Roussopoulos et al. (2017). "Smart Security Systems for Educational Institutions."
[3] Emerging Trends in ML-Based Security Systems, 2023.
[4] MILOUD BAGAA et al. (2017), A Machine Learning Security Framework for Iot Systems, May 21, 2020, IEEE Access, Digital Object Identifier 10.1109/ACCESS.2020.2996214
[5] Smart Threat Alert System using IoT, Dr. Bhawna Suri et all, International Conference on Computing, Communication and Automation (ICCCA2017), ISBN: 978-1-5090 6471-7/17/$31.00
[6] ©2017 IEEE
[7] IoT and Big Data Technologies: Opportunities and Challenges for Higher Learning Ruth Chweya et all, International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume- 9 Issue-2, July 2020
[8] BEHAVIOUR-BASED SECURITY WITH MACHINE LEARNING ON IOT NETWORKS,
[9] Kahraman Kostas, Department of Computer Science School of Mathematical and Computer Sciences Heriot-Watt University October 2023
[10] Using machine learning algorithms to enhance IoT system security, Hosam El-Sofany et all,
[11] Scientifc Reports | (2024) 14:12077 | https://doi.org/10.1038/s41598-024-62861-y
[12] Retracted: Towards Development of Machine Learning Framework for Enhancing Security in Internet of Things, Mutyalaiah Paricherla et all , Hindawi Security and Communication Networks Volume 2023, Article ID 9854892, 1 page https://doi.org/10.1155/2023/9854892, Received 10 October
[13] 2023; Accepted 10 October 2023; Published 11 October 2023,