Suspicious Activity Detection in Exam Hall

Vaishnavi V. Taware[1], Sakshi D. Raut[2], Ritika S. Jadhav[3], Suknya V. Late[4], Prof. Saba Chaugule[5]
[1,2,3,4]*Student, Department of Computer Engineering, P. K. Technical Campus, Pune*
[5]*Professor Department of Computer Engineering, P. K. Technical Campus, Pune*

***Abstract*—This project, titled "Suspicious Activity Detection in Exam Hall Using CNN," aims to automatically detect and alert exam supervisors about suspicious behavior during examinations. The system uses video input from a camera and processes it through image preprocessing and feature extraction techniques. A Convolutional Neural Network (CNN) model is trained using a dataset of normal and suspicious activities to identify unusual behavior such as cheating or using mobile phones. When suspicious activity is detected, the system captures the image and sends an email alert to the examiner. This project helps to ensure a fair and transparent examination process by reducing manual monitoring and improving accuracy and reliability through artificial intelligence.**

I. INTRODUCTION

Examination environment consists of large number of candidates and number of exam conductors and guards. Lot of human efforts are needed to continuously watch over the candidates for prevention of any mishap or cheating.

A human cannot always pay attention and concentration of this much to the exam. Hence, in order to address the problem of cheating in exam and for reduction in human efforts, this suspicious activity detection system is designed. In existing system, the detection of abnormal activity is done in a examination environment. That system uses 3D Convolution Neural Network (CNN) for detection which comprises of two convolution layers instead of five. System is able to detect only the basic gestures of the candidates in an examination environment. Existing system is not able to detect the facial features. Also, it is not able to recognize or identify the particular candidate based on his/her suspicious activity.

II. PROBLEM STATEMENT

During examinations, it is difficult for invigilators to continuously monitor all students and identify suspicious activities. Manual observation is time-consuming and prone to human error. To solve this problem, a smart system is needed that can automatically detect suspicious behavior using video analysis and send instant alerts to the examiner.

III. METHODOLOGY

The system captures live video from the exam hall using a camera. The video is divided into frames and goes through image preprocessing to remove noise and improve clarity. Then, using CNN, important features are extracted and classified as normal or suspicious activity. If normal, the system continues monitoring; if suspicious, it captures the image and sends an alert email to the examiner. This architecture helps in automatic monitoring, reduces manual work, and ensures a fair and secure examination process

IV. MACHINE LEARNING MODELS

A. Convolutional Neural Network (CNN):

The Suspicious Activity Detection system primarily utilizes a Convolutional Neural Network (CNN), a deep learning-based approach widely used for image analysis and pattern recognition. CNNs are particularly effective in extracting spatial hierarchies of features from images, making them suitable for identifying complex human behaviors in real-time video streams.

In this system, continuous video input from surveillance cameras installed in the examination hall is captured and segmented into individual frames. Each frame undergoes preprocessing steps such as resizing to a fixed dimension, normalization of pixel

values, and noise reduction to ensure consistency and improve model performance. The pre-processed frames are then passed through multiple layers of the CNN architecture, including convolutional layers, activation functions (such as ReLU), pooling layers, and fully connected layers. The convolutional layers extract low-level features such as edges and textures, while deeper layers capture high-level features such as human posture, gestures, and object interactions. The model is trained using a labeled dataset consisting of normal and suspicious activities, such as cheating gestures, use of mobile phones, or unusual movements. During training, backpropagation and optimization algorithms like stochastic gradient descent are used to minimize classification error. In the deployment phase, the trained CNN model performs real-time classification of incoming frames. If a frame is classified as suspicious, the system triggers an automated response which includes capturing the frame, storing it as evidence, and sending an alert notification via email to the exam supervisor.

The CNN-based approach provides high accuracy, robustness to variations in lighting and viewing angles, and the ability to generalize well to unseen scenarios. However, it requires significant computational resources and a large amount of training data for optimal performance.

B. Haar cascade Algorithm:

The Haar Cascade algorithm is a classical machine learning-based object detection technique that can also be used for suspicious activity detection in examination environments. It is particularly known for its fast detection speed and low computational requirements.

In this approach, the input video stream is first converted into grayscale for-mat to reduce computational complexity. The Haar Cascade classifier then processes each frame using a sliding window mechanism to detect predefined features. The detection is based on Haar-like features, which are simple rectangular patterns rep-resenting differences in pixel intensity between adjacent regions. The algorithm uses an integral image representation to quickly compute these features and employs a cascade of classifiers trained using the AdaBoost algorithm. This cascade structure allows the system to discard non-relevant regions quickly while focusing computational effort on promising areas of the image. Pre-trained classifiers are used to detect objects such as faces, eyes, and potentially mobile phones. By analyzing the position and movement of these detected features across frames, the system can infer suspicious behaviors such as frequent head turning, looking away from the examination sheet, or the presence of unauthorized objects. When such behavior exceeds a predefined threshold, the system flags it as suspicious, captures the corresponding frame, and sends an alert notification to the examiner. Although the Haar Cascade method is efficient and suitable for real-time ap-plications on systems with limited resources, it has limitations in terms of accuracy and robustness. It is sensitive to changes in lighting conditions, object orientation, and occlusions, making it less reliable compared to deep learning approaches like CNN for complex behavior analysis.

C. Software Requirements

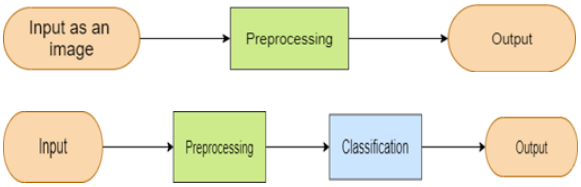
Operating System: Windows 10
IDE: PyCharm, Spyder
Programming Language: Python
Libraries: OpenCV, TensorFlow/Keras, NumPy, Pandas

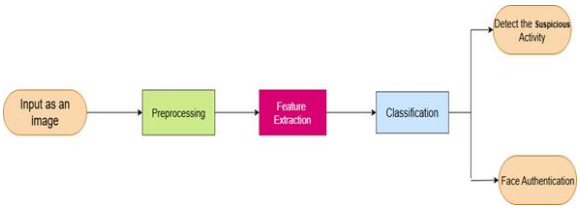
D. Hardware Requirements

- System Processors: Core i3 or higher
- Speed: 2.4 GHz or above
- Hard Disk: 150 GB or more
- Camera: CCTV or Web Camera for video input

V. DATA FLOW DIAGRAM

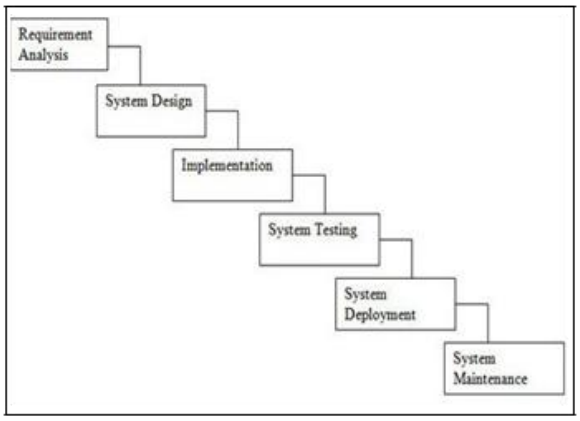
In Data Flow Diagram, we Show that flow of data in our system in DFD0 we show that base DFD in which rectangle present input as well as output and circle show our system, In DFD1 we show actual input and actual output of system input of our system is text or image and output is rum or detected likewise in DFD 2 we present operation of user as well as admin





VI. ENTITY RELATIONSHIP DIAGRAM

VII. PROJECT ESTIMATE



1) Requirement gathering and analysis:
In this step of waterfall, we identify what are various requirements are need for our project such are software and hardware required, database, and interfaces.

2) System Design:
In this system design phase, we design the system which is easily understood for end user i.e. user friendly.

We design some UML diagrams and data flow diagram to understand the system flow and system module and sequence of execution.

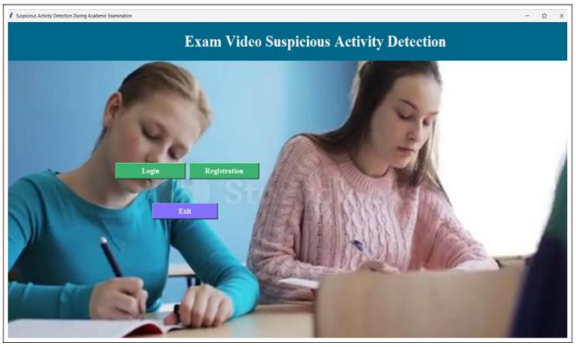
3) Implementation:
In implementation phase of our project, we have implemented various module required of successfully getting expected outcome at the different module levels.
With inputs from system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality which is referred to as Unit Testing.

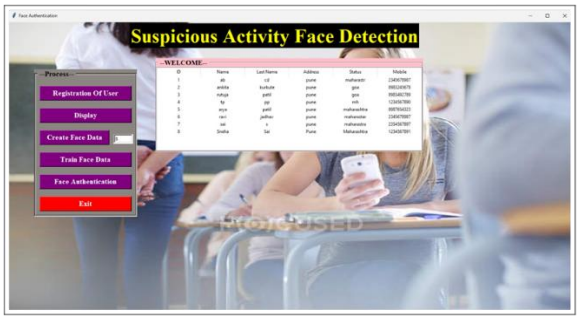
4) Testing:
The different test cases are performed to test whether the project module is giving expected outcome in assumed time. All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.

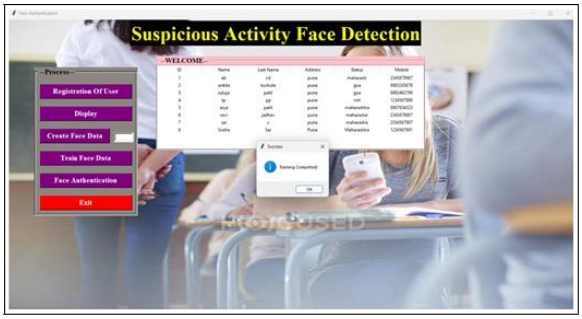
5)Deployment of System:
Once the functional and non-functional testing is done, the product is deployed in the customer environment or released into the market.

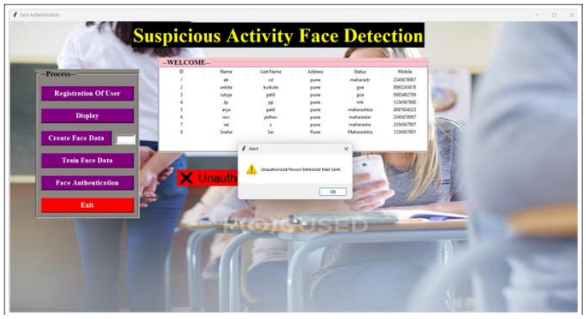
6)Maintenance:
There are some issues which come up in the client environment. To fix those issues patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

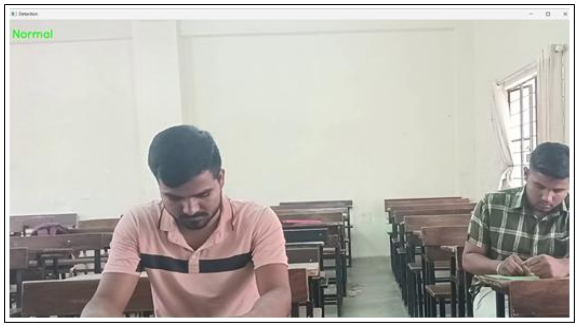
VIII. RESULTS











IX. APPLICATION

- Examination halls for automatic monitoring of students.
- Schools, colleges, and universities for maintaining academic integrity.
- Online and remote exams to prevent cheating.
- Corporate tests and recruitment assessments.
- Surveillance systems for detecting suspicious or abnormal behaviour.

- Public safety monitoring, such as in libraries or testing centres.

X. FUTURE SCOPE

- The system can be improved using advanced deep learning models for higher accuracy.
- It can include face recognition to identify individual students during exams.
- Voice detection can be added to detect talking or whispering.
- Can be connected to a mobile app or web dashboard for live monitoring.
- Cloud storage can be used to save and review suspicious activity recordings.
- The system can be upgraded to work in real-time across multiple cameras.
- Can be applied in other areas like surveillance, security, and crowd monitoring.

XI. CONCLUSION

The project "Suspicious Activity Detection in Exam Hall Using CNN" successfully provides an automated way to monitor students and detect unfair activities during examinations. The system uses video input from a camera and analyzes it through a Convolutional Neural Network (CNN) to identify suspicious movements or behavior. When such activity is detected, it captures the image and sends an email alert to the examiner. This system helps reduce manual supervision, saves time, and ensures a fair and transparent examination process. It is cost-effective, reliable, and efficient, making it useful for schools, colleges, and online exam centers. In the future, it can be enhanced with additional features like face and voice recognition for better accuracy and security.

ACKNOWLEDGMENT

This paper wouldn't be written without the respected advice of Prof SABA Chaugule. Our special thanks visit all the professors of the computer engineering department of P K Technical Campus for their support and for giving a chance to figure on a survey of software task extraction and navigation.

REFERENCES

[1] K. Rezaee, S. M. Rezakhani, M. R. Khosravi, and M. K. Moghimi, "A Survey on Deep Learning-Based Real-Time Crowd Anomaly Detection for Secure Distributed Video Surveillance," Personal and Ubiquitous Computing, vol. 28, no. 1, pp. 135–151, Feb. 2024.

[2] M. Perez, A. C. Kot, and A. Rocha, "Detection of Real-World Fights in Surveillance Videos," in Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), May 2019, pp. 2662–2666.

[3] C. V. Amrutha, C. Jyotsna, and J. Amudha, "Deep Learning Approach for Suspicious Activity Detection from Surveillance Video," in Proceedings of the 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), Mar. 2020, pp. 335–339.

[4] W. Sultani, C. Chen, and M. Shah, "Real-World Anomaly Detection in Surveillance Videos," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2018, pp. 6479–6488.

[5] J. Wei, J. Zhao, Y. Zhao, and Z. Zhao, "Unsupervised Anomaly Detection for Traffic Surveillance Based on Background Modeling," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Jun. 2018, pp. 129–136.