SUSPICIOUS ACTIVITY DETECTION DURING PHYSICAL EXAMS

Muhammad Asad
Department of Electrical Electronics and Communication Engineering
University of Engineering and Technology Lahore, Faisalabad Campus
Sheikhupura, Pakistan
2019ee383@student.uet.edu.pk

Mateen Abbas
Department of Electrical Electronics and Communication Engineering
University of Engineering and Technology Lahore, Faisalabad Campus
Gujranwala, Pakistan
abbasmubeen14@gmail.com

Asim
Department of Electrical Electronics and Communication Engineering
University of Engineering and Technology Lahore, Faisalabad Campus
Lahore,Pakistan
2019ee401@student.uet.edu.pk

Ahmed Hafeez
Department of Electrical Electronics and Communication Engineering
University of Engineering and Technology Lahore, Faisalabad Campus
Lahore,Pakistan
2019ee432@student.uet.edu.pk

Ms. Munazza Sadaf
Department of Electrical Electronics and Communication Engineering
University of Engineering and Technology Lahore, Faisalabad Campus
Faisalabad, Pakistan
munazzasadaf@uet.edu.pk

Muhammad Ahsan UL Haq
Department of Electrical Electronics and Communication Engineering
University of Engineering and Technology Lahore, Faisalabad Campus
Faisalabad, Pakistan
ahsanulhaq@uet.edu.pk

Muhammad Asif
Department of Electrical Electronics and Communication Engineering
University of Engineering and Technology Lahore, Faisalabad Campus
Lahore,Pakistan
emasif185@gmail.com

Abstract— ***Cheating in academic environments has become a significant problem, and traditional methods of detecting cheating are no longer adequate. Machine learning has the potential to detect cheating behavior that might go unnoticed by traditional methods and can also learn from the data it analyzes to continuously improve its performance. This research aims to develop a cheating management system that utilizes deep learning algorithms to detect and prevent cheating in academic environments. The proposed system analyzes video frames from cameras, to identify suspicious behavior and also provides instant feedback to the invigilators, and generates reports to assist in identifying patterns and trends in cheating behavior. The system utilizes convolutional neural networks (CNNs), which are particularly well-suited for image analysis, making them ideal for detecting cheating behavior. The proposed system is designed to work in real-time, providing instant feedback to the invigilators. The system also acts as a deterrent by making students aware that their behavior is being monitored. With the help of generated reports, this system identifies patterns and trends in cheating behavior, which can help educational institutions develop strategies to prevent cheating during physical exams. The accuracy of 0.90 is obtained with the help of Cub-SVM classifier. CSVM is overall the best and has better performance.***

Keywords— Suspicious activity, Ai invigilator, Deep learning, Convolutional neural networks, Computer vision

I. INTRODUCTION

Cheating has become a major concern for educational institutions around the world. With the advancement of technology, students have access to a vast amount of information, which has made it easier for them to cheat. Traditional methods of detecting cheating such as proctoring and invigilation have proven to be ineffective in some cases. Therefore, there is a need for a more effective cheating management system. This research is important because cheating destroys the integrity of the educational system and may waste the hard work of honest students. By developing a more effective cheating management system, we can ensure that students are evaluated fairly and that their hard work is recognized. This research will also contribute to the ongoing research in artificial intelligence or its application in cheating detection. Proposed system will be designed to work in real-time, providing instant feedback to the invigilators. The system will also generate reports that will assist in identifying patterns and trends in cheating behavior.

Reason of this project was to create a system that will not only detection of cheating but also prevent it by acting as a deterrent. This final year project aims to develop a cheating management system that can detect and prevent cheating in academic environments. This system will use machine learning to analyze data from various sources, providing instant feedback to the invigilators and generating reports to assist in identifying patterns and trends in cheating behavior. In recent times, deep learning has aroused as a promising tool to detect cheating, and this final year project aims to develop a cheating management system using deep learning algorithms. Deep learning is a part of Machine learning that is the part of Intelligence[1]. It is particularly well-suited for analyzing large amounts of data, which makes it ideal for detecting cheating behavior in academic environments.

The proposed system utilized deep learning algorithms to analyze data from cameras to identify suspicious behavior. The reason of this research is to utilize a cheating management system that is more effective than traditional methods of detecting cheating. By utilizing deep learning algorithms, in addition, the system will act as a deterrent by making students aware that their behavior is being monitored[2]. This research will analyze data from cameras, provide instant feedback to the invigilators, and generate reports to assist in identifying patterns and trends in cheating behavior. Cheating in educational settings is typically addressed at the classroom or institutional level. As humans, we possess a remarkable ability to perceive information conveyed through various movements, such as gestures or overall body motion. Human Activity Recognition (HAR) finds extensive utility in smart homes, serving two important purposes: care for disabled and elders, and to adopt the kind of residents-based environment and their behavior to enhance their quality of life[3]

In the present day, HAR have been using in many sectors where top concern is security. For example, airports utilize HAR to detect passenger behavior in areas like boardings, zones of baggage claim, areas of arrival, school entrances, and even canteens to identify instances of fighting, many lost objects, theft and vandalism etc.

Due to the COVID-19 pandemic, the demand for student dishonesty detection systems has increased significantly in universities. Ensuring the safety of invigilators and students alike, the detection of suspicious activities during exams has become crucial. Acts such as watching from behind, looking directly at others or carrying a cellphone in the examination halls require immediate attention. An intelligent project can give issuing warnings or alarms is needed to address these activities effectively. Accurate activity prediction can assist invigilators in taking appropriate actions, while also reducing potential biases. The impact of HAR extends beyond exams and holds great potential in various human activities.

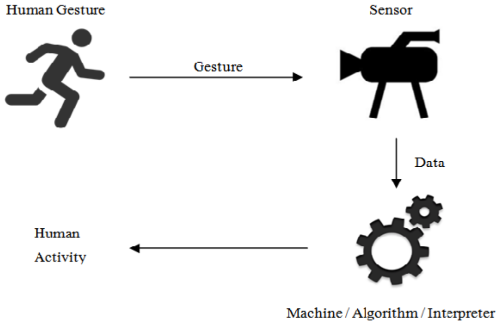


FIGURE 1: General structure of HAR system

Algorithms of deep learnings can automatically catch various actions, including:

(a)Front which may look normal watching at front, Front Right, Right, Back, Left, Front Left

(b)Cell Phones

Automating surveillance events can alleviate the workload of invigilators during exams. However, despite its significance, activity detection can be affected by several technical limitations.

The following challenges are commonly encountered:

(a) Occlusion, where objects obstruct the view.

(b) Variations of illumination

(c) Device size variations.

(d) Changes of appearances due to different clothing.

(e) Computational time constraints.

To address these challenges, this work presents a methodology for suspicious activity recognition based on deep feature extraction. 63-layer CNN algorithm, like L4-BranchedActionNet, was suggested for acquisition of features. Initially this is trained using the datasets of exams or their features for recognition of suspicious activities that was extracted from this pre-trained network. An entropy-coded ACS algorithm is used for the selection of the feature subsets, or many classifiers are used for the evaluation of their performance. Given 8results demonstrate the successful achievement of the intended objectives.

A. Problem Statement

Surveillance systems in instructional institutions often rely upon CCTV cameras to screen college students and maintain a comfy and truthful examination environment. However, manually monitoring the camera feeds and identifying suspicious behaviors is a hard and time-eating project for human operators. The present strategies for scholar conduct detection and class regularly contain guide statement or fundamental rule-based totally processes, that are subjective, susceptible to errors, and absence the capability to handle complex eventualities[4]. Moreover, these methods do now not leverage the advancements in deep learning and laptop vision techniques, which have proven excellent capacity in studying visible data and extracting meaningful capabilities. Consequently, there's a need for an automated machine that may appropriately stumble on and classify suspicious activities in actual-time from the surveillance camera feeds. Such a machine might assist make certain the integrity of exams, prevent dishonest, and provide well timed alerts for intervention whilst necessary. Moreover, the system has to be capable of handling diverse situations and variations in lights conditions, digicam angles, and scholar behaviors.

The number one goal of this research is to increase a strong and efficient framework that integrates deep studying-based characteristic extraction, function fusion, and category strategies to cope with the aforementioned problem. The framework goals to leverage the energy of deep gaining knowledge of fashions to mechanically analyze discriminative features from the input facts, fuse these capabilities to capture comprehensive statistics, and utilize suitable classifiers to as it should be classified and identified.

II. LITERATURE REVIEW

Cheating in academic environments has become a significant problem, and traditional methods of detecting cheating are no longer adequate. In recent times, there has been a growing interest in utilizing machine learning algorithms to detect cheating behavior. Machine learning has the potential to detect cheating behavior that might go unnoticed by traditional methods and can also learn from the data it analyzes to continuously improve its performance[5]. The utilization of machine learning algorithms to detect cheating in online exams utilized several machine-learning techniques, including decision trees, support vector machines and random forests. The results showed that these algorithms were effective in detecting cheating behavior, with an accuracy of over 90%. The study concluded that machine learning algorithms have the potential to detect cheating in online exams and can be used to develop effective cheating detection systems for physical exams[6]. The use of machine learning algorithms to detect cheating in programming exams. The study utilized machine learning techniques, including artificial neural networks and decision trees, to detect cheating behavior in programming exams. The results showed that these algorithms were effective in detecting cheating behavior, with an accuracy of over 90%[7]. Several studies have also investigated that deep learning algorithms, which are a subset of machine learning algorithms that can learn from large datasets[1]. The use of deep learning algorithms to detect cheating behavior in online exams. The study utilized convolutional neural networks (CNNs), a type of deep learning algorithm, to detect cheating behavior. The results showed that CNNs were effective in detecting cheating behavior, with an accuracy of over 95%. The study concluded that deep learning algorithms, such as CNNs, can be used to develop highly accurate cheating detection systems[8]. Deep learning algorithms to detect cheating behavior in multiple-choice exams. The study utilized a combination of CNNs and long short-term memory (LSTM) networks, another type of deep learning algorithm, to detect cheating behavior. The results showed that the proposed system was effective in detecting cheating behavior, with an accuracy of over 90%[9].

This shows that machine learning and deep learning algorithms have the potential to detect cheating behavior in academic environments. Several studies have investigated the use of machine learning and deep learning algorithms in detecting cheating behavior in various types of exams, such as online exams, physical exams[6]. The studies have shown that these algorithms are effective in detecting cheating behavior, with high levels of accuracy [8]. The use of machine learning and deep learning algorithms in cheating management systems is a promising area of research, and this project will contribute to the ongoing research in this field.

III. Methodology

This section offers a comprehensive evaluation of the materials and implementations utilized in this research. It includes an in depth clarification of the proposed 63-layer CNN model, which serves as the core of the framework. The framework comprises several vital steps, along with records training, hand labeling of the information, training the CNN structure using the custom dataset, extracting functions from the pre-trained dataset the usage of the proposed CNN version, appearing feature subset choice using the Ant Colony machine algorithm, and employing various classifiers for category. To facilitate self-sustaining characteristic extraction and classification, a singular CNN-based model referred to as L4-Branched-ActionNet, which includes sixty three layers, is brought. Moreover, the chosen function vector is inputted into SVM-primarily based classifiers to achieve category consequences. Every step depicted within the block diagram might be elaborated on inside the subsequent section. Accurately labeling students' suspicious activities at some point of the exam is of utmost importance in this context. The number one objective of this has a look at is to advise a CNN architecture for the detection of suspicious activities throughout examinations, which necessitates specific labeling of handcrafted records. The framework accommodates several key steps, along with statistics coaching, handmade records labeling, training the proposed Convolutional Neural community (CNN) structure using the custom dataset, extracting capabilities from the action recognition dataset the usage of the proposed Convolutional Neural network (CNN) architecture, performing characteristic subset choice with the Ant Colony system (ACS) set of rules, and employing various classifiers for classification.

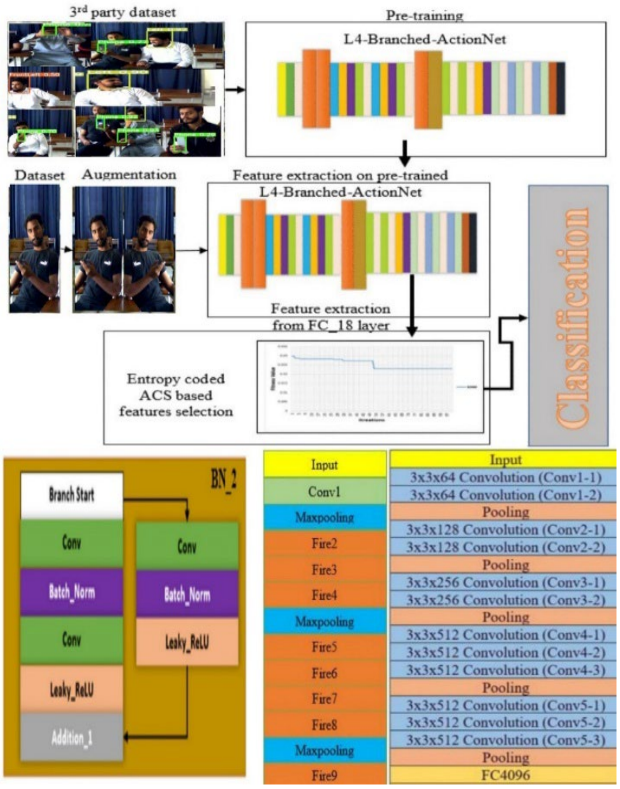


FIGURE 2: Structure of proposed frame work (The intuition of the proposed L4-Branched-ActionNet based framework for suspicious activity recognition)

To accomplish this objective, we endorse a novel CNN-primarily based version with 63 layers referred to as L4-Branched-ActionNet, which helps as the inspiration of suggested architecture pipeline. The Visual Geometry Group (VGG)-16 network is used as the baseline, and the filters are divided into multiple groups in a Convolutional operation[10].

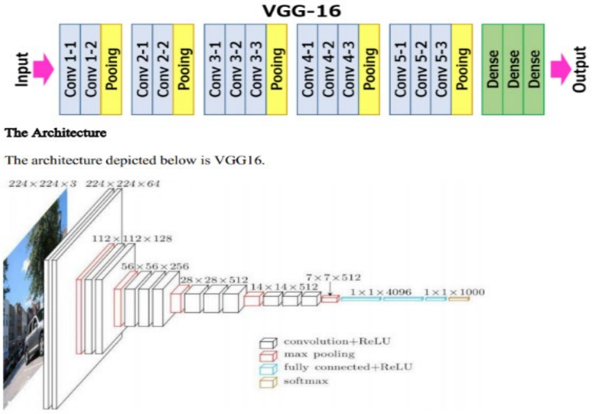


FIGURE 3: VGG-16 is a convolutional neural network that is 16 layers deep

Each group oversees a set of 2D convolutions within a specific range. Batch Normalization is applied to adjust channel neurons based on a defined batch size. It calculates the mean and variance within batches, derives the mean, and separates the features using the standard deviation.

$$Mean_B = \frac{1}{\omega}\sum_{z=1}^{w} \| z$$

Where w is the total number of features that are present in a single batch. And Variance can be calculated as follows:

$$Variance_B = \frac{1}{\omega}\sum_{z=1}^{w} (\mathrm{I}z - Mean_B)^2$$

The CNN model proposed in this study integrates both ReLU and Leaky ReLU operations. ReLU transforms all values below 0 to 0, while Leaky ReLU, instead of being 0, has a small slope for negative values. This small slope in Leaky ReLU helps in preventing complete saturation and allows for a more gradual transition for negative values[11].

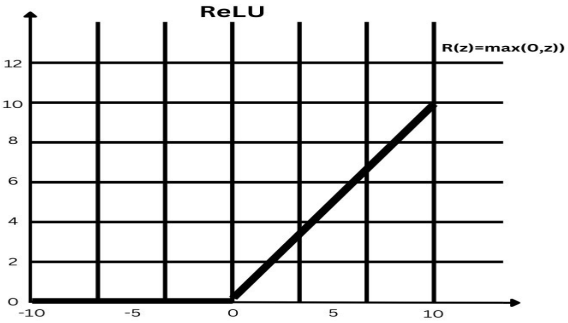


FIGURE 4: Small Slope ReLU

The 1st step in the proposed framework is data preparation, which involves collecting and organizing the dataset. In this work, we used the custom dataset, which contains 2500 images which are segments of videos recorded for that purpose of seven different classes (Front, Front Right, Right, Back, Left, Front Left, Phone). The video clips are captured at 640x480 resolution and a frame rate of 30 fps and converted into 2500 frames that were used in the training.

The second step is handcrafted data labeling. The most important for this process is the labelling of students. The labeled dataset is then broken into the testing and training sets. We followed the empirical studies and divided our data in which 75% for training and 25% was for the testing the architecture[12]. Training sets are used for the training of proposed CNN architecture, while the testing is used for evaluation of performance.

Next step is towards the training of CNN architecture on the dataset of given custom dataset[13]. The given 63 layers L4-Branched-ActionNet CNN network is designed after some studies[14]. Multiple approaches have been used in finalizing given architecture. Most important approaches are fine- adding, tuning, and removal of different layers. The training is aided with Intel HD Graphics 620 GPU comprising 4 GB V-RAM.

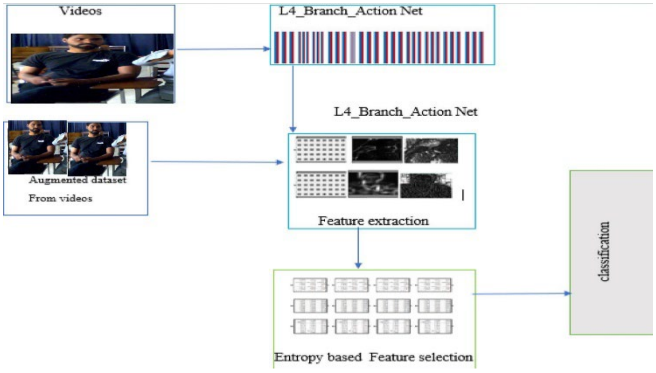


FIGURE 5: Proposed Deep L4-BranchedActionNet CNN architecture

The fourth step is features extraction of the motion recognition dataset on the proposed CNN structure. The CNN architecture is used to extract capabilities following feature choice. The pre-education is made on custom dataset.

The fifth step is features subset selection using the ACS algorithm. The ACS algorithm is used to select the most discriminative features from the extracted feature set. The final step is classification using various classifiers[15]. The ensembled selected feature vector is fed to the SVM-based classifiers to get the class outcomes. The overall performance of the proposed framework is evaluated primarily based on exceptional overall performance metrics, along with accuracy, precision and F-measure. In conclusion, the proposed framework is a novel and effective approach for detecting suspicious activities during the examination. The use of a CNN-based model with 63 layers and the ACS algorithm for feature subset selection has shown promising results in terms of accuracy and performance metrics[16]. This research has contributed to the field of human activity recognition.

While ensuring human safety and health, the surveillance security of HAR has garnered significant attention from researchers. L4-Branched-ActionNet become mostly added to facilitate the education manner throughout clustered GPUs with restrained memory capacity[17]. On this structure, the filters are divided into multiple groups inside a Convolutional operation, in which each institution oversees a set of second convolutions within a selected range. To adjust channel neurons over a defined batch length, Batch Normalization is hired, which calculates the suggest and variance inside fragments. The mean is then derived, and the functions are separated the use of the standard deviation.

This study presents a framework dedicated to classifying suspicious activities exhibited by students in an examination hall. The activities are labeled based on input acquired from surveillance cameras deployed during the examination. The proposed approach is specifically designed as a computer vision-based system.

The methodology encompasses several key steps. Firstly, the images in the complete dataset are resized and converted into grayscale[18]. Ultimately, feature extraction, choice, and class of the photographs are accomplished. The fused capabilities are then subjected to function choice the usage of the major factor evaluation (PCA) approach. Ultimately, the chosen functions go through category the use of help Vector device (SVM) and nice k-Nearest pals (KNN) algorithms. To evaluate the efficacy of the proposed method, an external dataset is hired.

The techniques used in this research are aimed at providing a framework for classifying suspicious activities of students during checks the usage of pc imaginative and prescient. The proposed methodology integrates photo resizing, grayscale conversion, function extraction, feature selection, and category of pictures.

Data Preparation: The dataset used in this study consists of video footage captured by surveillance cameras during exams. The videos are processed and converted into grayscale images, resulting in a set of images for each video. The images are then resized to 224 x 224 pixels to fit the input size of the proposed CNN architecture[11], [18].

Handcrafted Data Labeling: The images in the dataset are labeled according to the suspicious activity they depict. The labeling is done manually, and the seven classes of suspicious activity are: Front, Front Right, Right, Back, Left, Front Left, Phone[12], [13].

The CNN architecture proposed in this research is trained on the custom dataset to extract highly informative features through a feature selection process. The development of the 63-layer Deep L4-Branched-ActionNet CNN Network involved a series of comprehensive experiments. Various techniques, such as fine-tuning, layer addition, and layer removal, were explored to optimize the architecture[19]. After extensive evaluation, the final form of the architecture with 63 layers was determined to yield superior performance and deliver the best outcomes.

Feature Extraction: After the pre-training is complete, the proposed CNN architecture is used to extract features from the exam surveillance dataset. The CNN architecture is designed to extract rich and powerful features from images, which can then be used to classify the suspicious activity depicted in the images[20].

Feature Subset Selection: The extracted features are then subjected to feature subset selection using the Ant Colony Search algorithm. This algorithm selects a subset of the most informative features that contribute to the classification performance of the model[21].

Classification: The feature subset that has been selected is then utilized as input for the support vector machine (SVM) and k-nearest neighbor (KNN) classifiers to classify the suspicious activities. SVM and KNN classifiers are chosen for their ability to handle high-dimensional data with high accuracy and robustness. Performance evaluation of the classifiers is conducted using metrics such as accuracy, precision, recall, and F1-score[9], [15].

Principal Component Analysis: To lower dimensionality features that are extracted, Principal Component Analysis (PCA) is employed. PCA is a statistical technique for the transforms the original features into a lower-dimensional space while preserving the most critical information[22].

The effectiveness of the proposed methodology is assessed using external dataset, and the system's performance is analyzed in terms of precision, accuracy and F-measure. The results demonstrate that the proposed framework achieves a high level of accuracy in classifying suspicious activities displayed by students during exams.

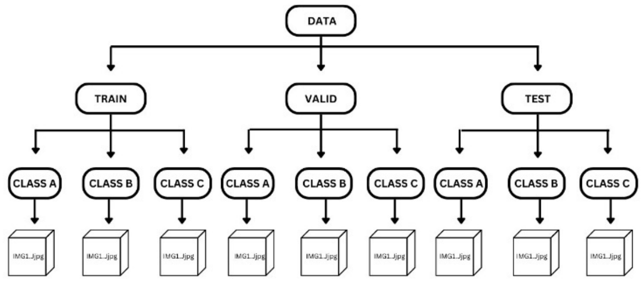


FIGURE 6: Dataset Description

A. ACS FEATURE SUBSET SELECTION

This approach falls within the wrapper-based approach, also referred to as ant colony system-based feature optimization, which utilizes principles from probability theory and draws inspiration from the behavior of ants. When ants travel, they release a substance called pheromone, which gradually diminishes over time. The ants choose their path by probabilistically favoring routes with higher pheromone intensity, enabling them to find the most efficient route with the least cost. This process resembles the movement of ants traversing from one node to another in a graph.

Ant colony optimization-primarily based function subset selection is a wrapper-based totally approach inspired by means of the foraging behavior of ants. The set of rules seeks to pick the most excellent functions that could cause the fine overall performance of the classification model. It is primarily based at the chance idea and is known as ant colony system-based characteristic optimization[23].

In this approach, the features are conceptualized as nodes in a graph, where the edges between nodes indicate the potential for selecting subsequent features. All nodes in the graph are interconnected in a mesh-like structure. Pheromone values are associated with each node (feature), and the algorithm mimics the behavior of ants to determine the optimal features. During foraging, ants release a substance called pheromone, which diminishes in intensity over time. Ants navigate by probabilistically selecting routes with higher pheromone concentrations. This mechanism guides them towards finding the path with the lowest cost[23], [24].

Similarly, the algorithm starts with random feature selection and iteratively evaluates the feature subset by training the model with selected features. The pheromone values are updated based on the performance of the classification model. By the dependence of probability at a mathematically selected feature

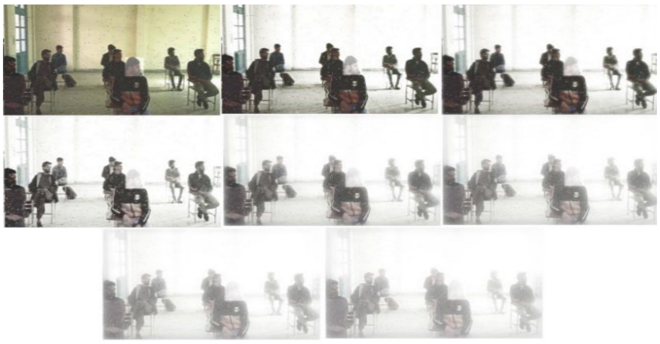
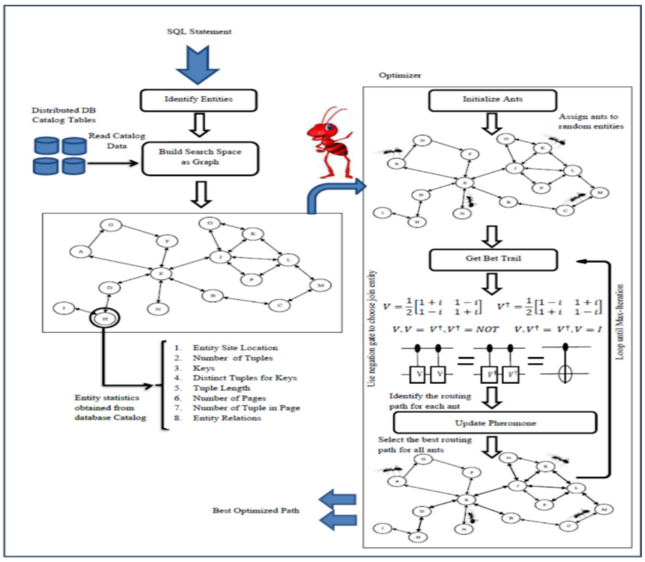


FIGURE 7: Image visualization of feature maps at various convolutional layers (a) Conv-1 (b) Conv-2 (c) Conv-3 (d) Conv-5 (e) G-Conv-8 (f) Conv-10 (g) Conv-12 (h) Conv-15 (i) G-Conv-17

$$Pij = τij^α * ηij^β / ∑ (τik^α * ηik^β)$$

Wherein Pij is the threat of choosing the function j at the i-th iteration, τij is the pheromone fee related to function j at the i-th iteration, ηij is the heuristic fee of function j on the i-th technology, α and β are the parameters that control the relative significance of pheromone and heuristic statistics and index of the chosen function subset[25]. The algorithm stops when the stopping condition is reached, such as when a minimum number of nodes are visited. Finally, the optimal feature subset is selected, and the classification model is trained with these features.

The ant colony optimization-based feature subset selection is a wrapper-based totally approach that simulates the behavior of ants in foraging to pick out the premier functions for the class version. Its miles based totally on the opportunity concept and makes use of pheromone values and heuristic information to choose the functions. The method is iterative and prevents whilst the preventing condition is reached. It is an effective feature choice technique which can enhance the overall performance of type fashions by selecting the optimal feature subset[26]. Each node in the graph represents a specific function, even as the rims among nodes indicate the possibility of selecting the following feature. The objective of the algorithm is to discover the ultimate set of functions. The set of rules terminates as soon as the minimum quantity of nodes has been visited and a predetermined stopping condition is met[24], [25], [27]. The nodes in the graph are interconnected in a mesh-like structure.


SQL Statement
Identify Entities
Distributed DB Catalog Tables
Read Catalog Data
Build Search Space as Graph
Optimizer
Initialize Ants
Assign ants to random entities
Get Bet Trail
$V = \frac{1}{2}\begin{bmatrix}1+i & 1-i\\1-i & 1+i\end{bmatrix}$ $V^\dagger = \frac{1}{2}\begin{bmatrix}1-i & 1+i\\1+i & 1-i\end{bmatrix}$
$V.V = V^\dagger.V^\dagger = NOT$ $V.V^\dagger = V^\dagger.V = I$
Use negation gate to choose join entity
Loop until Max-Iteration
Identify the routing path for each ant
Update Pheromone
Select the best routing path for all ants
1. Entity Site Location
2. Number of Tuples
3. Keys
4. Distinct Tuples for Keys
5. Tuple Length
6. Number of Pages
7. Number of Tuple in Page
8. Entity Relations
Entity statistics obtained from database Catalog
Best Optimized Path
FIGURE 8: Structure of Ant Colony System (ACS)[28]

B. FEATURE FUSION

Feature fusion is a technique used in machine learning to combine multiple features extracted from a dataset into a single, more informative feature vector. This process helps to reduce the noise in the feature space and highlight important information that may be hidden in the original features[20]. Feature fusion can be achieved through various techniques, such as concatenation, weighting, or selection.

One approach to feature fusion is concatenation, which involves combining the feature vectors horizontally into a single vector. For instance, if we have three feature vectors of sizes S1×d, L1×c, and G1×r, the concatenated feature vector will be of size R1×j, where j = d + c + r. This combined feature vector can then be used as input to a machine learning algorithm.

Another approach to feature fusion is weighting, which assigns different weights to the features based on their relative importance. In this approach, each feature vector is multiplied by a weight factor before being combined with the other feature vectors[29]. The weights can be learned using a training dataset or assigned manually based on prior knowledge of the importance of each feature.

Feature selection is a widely employed technique in feature fusion, aimed at choosing the most significant features from the original feature set. This manner contributes to lowering the dimensionality of the feature space and enhancing the performance of system getting to know ML algorithms[26]. Multiple approaches can be employed for feature selection, including principal component analysis

(PCA), mutual information, and recursive feature elimination (RFE).

Once the features have been combined, the resultant feature vector can be further processed using various machine learning algorithms. For instance, the Euclidean distance can be calculated for each feature with respect to the suggest value and placed through an activation function so one can arrange the vector within the least distance order[30]. This process helps to further refine the feature vector and reduce the impact of noisy features.

Feature fusion is an important technique used in machine learning to combine multiple features extracted from a dataset into a single, more informative feature vector. This process can help to reduce the noise in the feature space and highlight important information that may be hidden in the original features. Feature fusion can be achieved through various techniques, such as concatenation, weighting, or selection. The resultant feature vector can then be further processed using various machine learning algorithms to improve the accuracy of the predictions.

SFTA feature are the vectors given as:

$$F_{S1\times d} = \{S_{1\times1,}S_{1\times2,}S_{1\times3,} \dots S_{1\times d,}\}$$

Where $FS1\times d$ represents dimensions of SFTA, Furthermore Local Binary Patterns (LBP) feature is:

$$F_{L1\times c} = \{L_{1\times1,}L_{1\times2,}L_{1\times3,} \dots S_{1\times c,}\}$$

Where $FL1\times d$ represents dimensions of Local Binary Patterns (LBP) feature vector, Gabor feature vectors are:

$$F_{G1\times r} = \{G_{1\times1,}G_{1\times2,}G_{1\times3,} \dots G_{1\times r,}\}$$

Where $FG1\times d$ represents dimensions of the Gabor vector, the given input vectors are linked together successively or horizontally. The simplified vector is represented as:

$$F_{R1\times j} = \sum (F_{S1\times d}, F_{L1\times c}, F_{G1\times r})$$

Where $FR1\times j$ is output resultant of combined feature vector after simplification and j = d+ c + r then the mean is calculated of the output feature vector as:

$$F_k = \frac{1}{j}\sum F_{Ri}$$

After calculating the Euclidean distance of each feature from its mean value, an activation function is applied to arrange the vector in ascending order based on the distance. The Euclidean distances of the features are described as follows:$F_d = \{d_1, d_2, d_3, \dots d_j\}$. The resultant features vector is:

$$f_{F1\times k} = \{F_{1\times1,}F_{1\times2,}F_{1\times3,} \dots F_{1\times k,}\}$$

C. CLASSIFICATION

After the capabilities have been extracted, the system proceeds to the class section wherein diverse classifiers are employed, and a verification manner based on the chosen capabilities is integrated. In this section, the sports are categorized the usage of fine KNN and Cubic SVM classifiers. The capabilities, encoded with entropy-based definitely ACS, are surpassed to the predictor for categorization. A couple of variations of SVM and KNN classifiers, together with Linear SVM, Quadratic SVM, quality Gaussian SVM, Medium Gaussian SVM, Coarse Gaussian SVM, Cubic SVM, Cosine KNN, Coarse KNN, and best KNN, are carried out to assess the machine's ordinary overall performance.

After analyzing the performance outcomes, it is evident that Cubic SVM stands out as the most successful classifier for the chosen action dataset[26]. An intensive assessment become carried out in MATLAB, encompassing various class algorithms such as discriminant evaluation, Bayesian strategies, neural networks, assist vector machines, selection bushes, rule-primarily based classifiers, and nearest friends. The evaluation process involved custom datasets from the complete custom dataset to derive significant insights into the classifiers' behavior. Among all the classifiers, the SVM versions implemented and was demonstrated as the highest accuracy, reaching a maximum value of 0.90[15], [31].

Classification is an essential process in machine learning where objects or data are grouped into predefined categories or classes based on their features. It involves using algorithms and statistical models to analyze data and assign them to specific classes based on their characteristics.

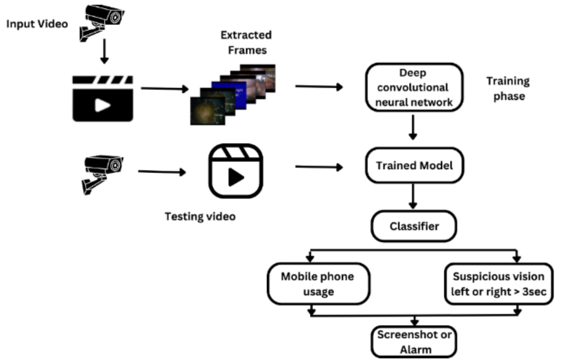


FIGURE 9: An Overview of complete system

The classification process usually involves two main stages: feature extraction and classification. Feature extraction is the process of identifying and extracting relevant features or attributes from the data. These features are then used as input variables for the classification algorithms. There are many feature extraction techniques available, depending on the nature of the data and the problem at hand. Some common techniques include statistical features, wavelet analysis, and Fourier transforms.

Once the features have been extracted, the next step is to classify the data into pre-defined categories. This is done by using a classifier, which is a mathematical algorithm that assigns objects to specific classes based on their features[14]. There are many different types of classifiers available, ranging from simple decision trees to complex neural networks.

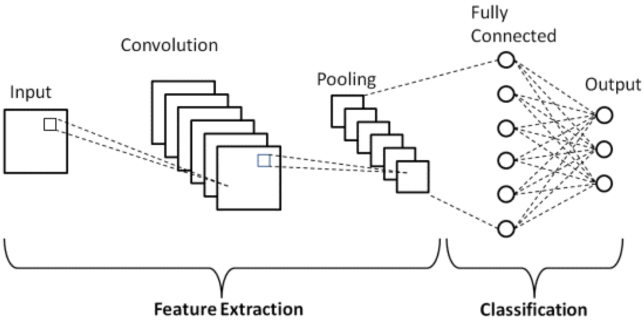


FIGURE 10: Image classification and modeling based on deep convolution neural network

The experiment assessments are repeated with the fine-taken into consideration configuration (a thousand functions using entropy-coded ACS function choice) at the outside take a look at dataset to take a look at the proposed framework's performance. The category procedure commonly entails major stages: characteristic extraction and category. Function extraction is the system of identifying and extracting relevant capabilities or attributes from the information. Those capabilities are then used as input variables for the category algorithms. There are many feature extraction techniques available, relying on the character of the facts and the trouble at hand. Some common techniques include statistical features, wavelet analysis, and Fourier transforms[26]. Once the features have been extracted, the next step is to classify the data into pre-defined categories. This is done by using a classifier, which is a mathematical algorithm that assigns objects to specific classes based on their features. There are many different types of classifiers available, ranging from simple decision trees to complex neural networks

Front	0.98	0	0	0.1	0	0.1	0
Front Right	0	1	0	0.01	0.1	0	0
Right	0	0	1	0	0	0	0
Back	0	0.08	0	0.88	0	0.07	0
Left	0.01	0	0.01	0	0.97	0	0.01
Front Left	0	0.01	0	0	0	0.99	0
Phone	0	0.01	0.01	0.01	0	0.01	0.96
	Front	Front Right	Right	Back	Left	Front Left	Phone

Table 1: Accuracy of algorithm assigning objects to specific classes based on their features

IV. RESULTS

The five-fold pass-validation approach is employed for each mastering and evaluation purposes. It includes randomly selecting 75% of the information for every fold to be used for training, even as the last 25% is reserved for testing. Diverse evaluation metrics are calculated from the output pictures, along with Accuracy, Confusion Matrix, Misclassification rate, occurrence, F-score, ROC Curves, and metrics including TPR, FPR, Sensitivity (SENSI), Specificity (SPECI), and Precision (PRECI), a good way to assess the algorithm's satisfactory and performance[32]. To generate the result or output image, the algorithm is applied to the provided input images obtained from the dataset for cheating detection in examinations. To validate the accuracy and efficiency of the algorithm, different calculations are performed on the evaluation metrics. Sensitivity and Specificity are evaluated to understand the algorithm's nature. High values of Sensitivity (SENSI) and Specificity (SPECI) indicate the effectiveness of the classifier and the overall quality of the algorithm. True Classification of the algorithm is ensured through the use of metrics like True Positive Rate (TPR) and True Negative Rate (TNR), which contribute to error-free and improved outputs. Metrics such as False Positive Rate (FPR) and False Negative Rate (FNR) are employed to quantify the Error Rate and validate the accuracy of the algorithm. The Receiver Operating Characteristic (ROC) curve represents the trade-off between Sensitivity (SENSI) and Specificity (SPECI) for a specific decision threshold, highlighting the differences between the classes. The Area Under the Curve (AUC) summarizes the model's performance in distinguishing between the classes[26].

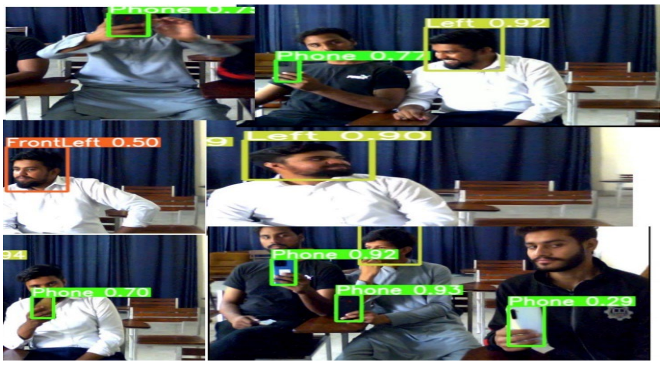


FIGURE 11: Results showing classifiers (Front, Front Right, Right, Back, Left, Front Left, Phone)

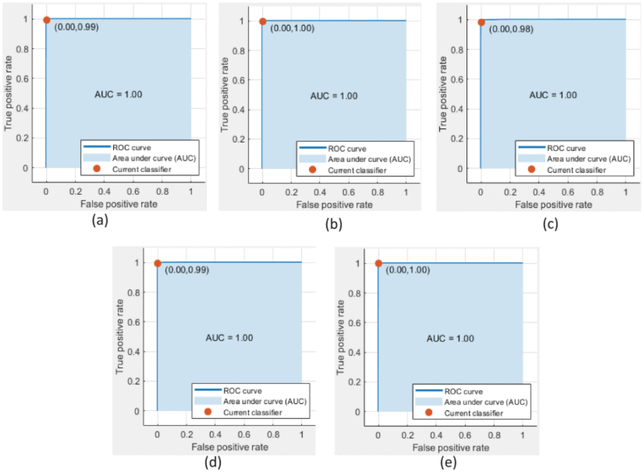


FIGURE 12: ROCs and AUCs of all classes having best results using CUB-SVM on custom dataset

$$Accuracy(Acu) = \frac{T_{Positives} + T_{negatives}}{T_{Positives} + T_{negatives} + F_{positives} + F_{negatives}}$$

$$Sensitivity\ (Sensi) = \frac{T_{Positives}}{T_{Positives} + F_{negatives}}$$

$$Specificity\ (Speci) = \frac{T_{negatives}}{T_{negatives} + F_{positives}}$$

$$AUC = \frac{T_{positives} + T_{negatives}}{T_{positives} + T_{negatives} + F_{positives} + F_{negatives}}$$

$$Precision(Preci) = \frac{T_{positives}}{T_{positives} + F_{positives}}$$

$$F_{measure}(FM) = 2 \times \frac{Preci \times Sensi}{Preci + Sensi}$$

$$G_{mean}\ (GM) = \sqrt{T_{positives-rate} \times T_{negatives-rate}}$$

Classifier	Acu(%)	Sensi(%)	Speci(%)	Preci(%)	FM(%)
Lin-SVM	0.891	1	0.876	0.6675	0.8991
Quad-SVM	0.912	1	0.95	0.78	0.883
Fin-SVM	0.53	0.701	0.612	0.198	0.223
Med-SVM	0.961	1	0.955	0.877	0.812
Cor-SVM	0.855	1	0.812	0.419	0.591
Cub-SVM	**0.98**	**1**	**1**	**0.88**	**0.97**
Cos-KNN	0.813	1	0.819	0.399	0.581
Cor-KNN	0.601	0.713	0.279	0.651	0.192
Fin-KNN	0.965	1	0.956	0.817	0.863

Table 2: Confusion matrix of the best outcome and Cub-SVM classifier on External Test dataset.

V. RECCOMENDATIONS

This study introduces a proposed architecture specifically designed to address the given dataset. The primary objective was to develop a CNN structure that effectively handles the dataset provided. The suggested architecture, called Deep L4-BranchedActionNet CNN Network, is designed for feature extraction after undergoing feature selection. Pretraining is performed on the external dataset, resulting in the derivation of a 63-layer Deep L4-Branched-ActionNet CNN Network through extensive experimentation. The architecture is refined based on experiments conducted on seven classes: Front, Front Right, Right, Back, Left, Front Left, Phone.

To further improve the proposed CNN architecture, it is recommended to test and evaluate it on larger and more diverse datasets. This will help assess its robustness and generalizability. Exploring alternative CNN architectures, such as ResNet and Inception, is also suggested to identify the optimal architecture for this task. Additionally, more attention should be given to the selection and preprocessing of the dataset. It is important to include more diverse and challenging scenarios to enhance its representativeness.

Considering alternative feature selection techniques like LDA and t-SNE, in addition to PCA, can provide deeper insights into the underlying data structure. Exploring data augmentation techniques, such as rotation, translation, and scaling, can enhance dataset diversity and improve model performance.

Employing ensemble methods is recommended to enhance model performance by combining multiple models and reducing overfitting.

While the proposed Deep L4-BranchedActionNet CNN architecture shows promise in detecting suspicious activities, further research is needed to fully realize its potential. Addressing the recommendations outlined above will help improve the architecture's performance and applicability in real-world scenarios.

VI. CONCLUSION

The purposed system has the ability to detect the suspicious activities that revolves around the academic area like monitoring student's activities during their physical exams. With the little bit updates this idea can be used in different scenarios like studying the behaviors of anyone at public or also in his/her private zone, in addition this can be used this to keep an eye on worker/trainers in industries where we can make sure that worker must obey the rules. This research describes the comprehensive framework for student's behavior detection and classification using deep learning-based feature extraction, feature fusion, and classification techniques.

The first step of this concept includes deep learning-based feature extraction approach that was utilized in order to retrieve the high-level information from the provided images. This includes the three types of features (SFTA, LBP, Gabor) that were extracted form custom dataset using CNN architecture. CNN design underwent refinement to withdraw relevant features for given task. The External dataset was utilized for processing by using both data augmentation and normalization techniques to improve the robustness of the model.

The second step was to fused the extracted features together to format a combined feature vector using this merging technique. Now perform normalization on the combined feature vector to make sure the equal importance of each feature during classification. In order to check the efficiency of the framework different classifiers like SVM and KNN algorithms were used to bind the feature vector.

In the field of research, monitoring the suspicious activities have much importance. With the help of this system, invigilators can perform legal actions against suspects and suspects/students will be automatically classified by our trained system. With the help of proposed 63 layered CNN Network named as L4-Branched-ActionNet the process of classifying the suspicious activities is enhanced. ACS scheme of entropy-coded are used to reduce the features. Different versions of SVM and KNN categorizers are processed in feature selection for the training and then testing of dataset. At the feature choice phase classifiers repeated the findings by modifying the count of features. The accuracy of 0.90 at lower performance is obtained on 100 features with the help of Cub-SVM classifier.

Using Cub-SVM classifier the best result having the accuracy of 0.90 with 1000 features. CSVM is overall best and have better performance. External dataset is used for the validation of the results and also comparison with recent outputs. The results that are acceptable and comparable, illustrate the validity of the recommended approach. Other CNN-Based Network can help for taking features in order to implement feature fusion. Our suggested system provides superior outcomes and this project will contribute to the ongoing research in this field.

References

[1] C. Janiesch, P. Zschech, and K. Heinrich, “Machine learning and deep learning,” *Electronic Markets*, vol. 31, no. 3, 2021, doi: 10.1007/s12525-021-00475-2.

[2] V. Singh, S. Singh, and P. Gupta, “Real-Time Anomaly Recognition Through CCTV Using Neural Networks,” in *Procedia Computer Science*, Elsevier B.V., 2020, pp. 254–263. doi: 10.1016/j.procs.2020.06.030.

[3] X. Li, Y. He, and X. Jing, “A survey of deep learning-based human activity recognition in radar,” *Remote Sensing*, vol. 11, no. 9. MDPI AG, May 01, 2019. doi: 10.3390/rs11091068.

[4] F. G. Filip, “AI vs AI (Augmenting [Human] Intellect vs Artificial Intelligence) : Plenary Talk,” in *SACI 2021 - IEEE 15th International Symposium on Applied Computational Intelligence and Informatics, Proceedings*, 2021. doi: 10.1109/SACI51354.2021.9465578.

[5] S. Grigorescu, B. Trasnea, T. Cocias, and G. Macesanu, “A survey of deep learning techniques for autonomous driving,” *J Field Robot*, vol. 37, no. 3, 2020, doi: 10.1002/rob.21918.

[6] A. S. Kulkarni, E. Naresh, M. Swetha, and S. M. Kusuma, “Automated System for Detection of Suspicious Activity in Examination Hall,” in *Proceedings of CONECCT 2021: 7th IEEE International Conference on Electronics, Computing and Communication Technologies*, Institute of Electrical and Electronics Engineers Inc., 2021. doi: 10.1109/CONECCT52877.2021.9622599.

[7] J. A. Ruiperez-Valiente, P. J. Munoz-Merino, G. Alexandron, and D. E. Pritchard, “Using Machine Learning to Detect ‘Multiple-Account’ Cheating and Analyze the Influence of Student and Problem Features,” *IEEE Transactions on Learning Technologies*, vol. 12, no. 1, 2019, doi: 10.1109/TLT.2017.2784420.

[8] R. Xin, J. Zhang, and Y. Shao, “Complex network classification with convolutional neural network,” *Tsinghua Sci Technol*, vol. 25, no. 4, pp. 447–457, Aug. 2020, doi: 10.26599/TST.2019.9010055.

[9] S. Essahraui, M. A. El Mrabet, M. F. Bouami, K. El Makkaoui, and A. Faize, “An Intelligent Anti-cheating Model in Education Exams,” in *Proceedings - 2022 5th International Conference on Advanced Communication Technologies and Networking, CommNet 2022*, 2022. doi: 10.1109/CommNet56067.2022.9993953.

[10] A. M. Ismael and A. Şengür, “Deep learning approaches for COVID-19 detection based on chest X-ray images,” *Expert Syst Appl*, vol. 164, 2021, doi: 10.1016/j.eswa.2020.114054.

[11] C. Dong, C. C. Loy, K. He, and X. Tang, “Image Super-Resolution Using Deep Convolutional Networks,” *IEEE Trans Pattern Anal Mach Intell*, vol. 38, no. 2, pp. 295–307, Feb. 2016, doi: 10.1109/TPAMI.2015.2439281.

[12] A. Diete, T. Sztyler, and H. Stuckenschmidt, “Exploring semi-supervised methods for labeling support in multimodal datasets,” *Sensors (Switzerland)*, vol. 18, no. 8, 2018, doi: 10.3390/s18082639.

[13] S. Athlur, N. Saran, M. Sivathanu, R. Ramjee, and N. Kwatra, “Varuna: Scalable, Low-cost Training of Massive Deep Learning Models,” in *EuroSys 2022 - Proceedings of the 17th European Conference on Computer Systems*, 2022. doi: 10.1145/3492321.3519584.

[14] R. Xin, J. Zhang, and Y. Shao, “Complex Network Classification with Convolutional Neural Network,” 2020. [Online]. Available: https://unstats.un.org/unsd/trade/sitcrev4.htm

[15] M. M. Krishna, M. Neelima, M. Harshali, and M. V. G. Rao, “Image classification using Deep learning,” *International Journal of Engineering and Technology(UAE)*, vol. 7, 2018, doi: 10.14419/ijet.v7i2.7.10892.

[16] S. S. Begampure and P. M. Jadhav, “Intelligent Video Analytics For Human Action Detection: A Deep Learning Approach With Transfer Learning,” *International Journal of Computing and Digital Systems*, vol. 11, no. 1, 2022, doi: 10.12785/ijcds/110105.

[17] M. D. Genemo, “Suspicious activity recognition for monitoring cheating in exams,” *Proceedings of the Indian National Science Academy*, vol. 88, no. 1. Springer Nature, Mar. 01, 2022. doi: 10.1007/s43538-022-00069-2.

[18] R. Zhang, P. Isola, and A. A. Efros, “Colorful image colorization,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2016. doi: 10.1007/978-3-319-46487-9_40.

[19] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, “A survey of the recent architectures of deep convolutional neural networks,” *Artif Intell Rev*, vol. 53, no. 8, 2020, doi: 10.1007/s10462-020-09825-6.

[20] K. N. Rode and R. J. Siddamallaiah, “Image Segmentation with Priority Based Apposite Feature Extraction Model for Detection of Multiple Sclerosis in MR Images Using Deep Learning Technique,” *Traitement du Signal*, vol. 39, no. 2, 2022, doi: 10.18280/ts.390242.

[21] S. Trabelsi, D. Samai, F. Dornaika, A. Benlamoudi, K. Bensid, and A. Taleb-Ahmed, “Efficient palmprint biometric identification systems using deep learning and feature selection methods,” *Neural Comput Appl*, vol. 34, no. 14, 2022, doi: 10.1007/s00521-022-07098-4.

[22] T. Li, J. Zhang, and Y. Zhang, “Classification of hyperspectral image based on deep belief networks,” in *2014 IEEE International Conference on Image Processing, ICIP 2014*, 2014. doi: 10.1109/ICIP.2014.7026039.

[23] M. Dorigo, V. Maniezzo, and A. Colorni, “Ant system: Optimization by a colony of cooperating agents,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 26, no. 1, 1996, doi: 10.1109/3477.484436.

[24] M. Dorigo and L. M. Gambardella, “Ant colony system: A cooperative learning approach to the traveling salesman problem,” *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, 1997, doi: 10.1109/4235.585892.

[25] X. Zhao, D. Li, B. Yang, C. Ma, Y. Zhu, and H. Chen, “Feature selection based on improved ant colony optimization for online detection of foreign fiber in cotton,” *Applied Soft Computing Journal*, vol. 24, 2014, doi: 10.1016/j.asoc.2014.07.024.

[26] T. Saba, A. Rehman, R. Latif, S. M. Fati, M. Raza, and M. Sharif, “Suspicious Activity Recognition Using Proposed Deep L4-Branched-Actionnet with Entropy Coded Ant Colony System Optimization,” *IEEE Access*, vol. 9, pp. 89181–89197, 2021, doi: 10.1109/ACCESS.2021.3091081.

[27] T. Saba, A. Rehman, R. Latif, S. M. Fati, M. Raza, and M. Sharif, “Suspicious Activity Recognition Using Proposed Deep L4-Branched-Actionnet with Entropy Coded Ant Colony System Optimization,” *IEEE Access*, vol. 9, pp. 89181–89197, 2021, doi: 10.1109/ACCESS.2021.3091081.

[28] S. A. Mohsin, A. Younes, and S. M. Darwish, “Dynamic cost ant colony algorithm to optimize query for distributed database based on quantum-inspired approach,” *Symmetry (Basel)*, vol. 13, no. 1, pp. 1–20, Jan. 2021, doi: 10.3390/sym13010070.

[29] G. Jia, H. K. Lam, and K. Althoefer, “Variable weight algorithm for convolutional neural networks and its applications to classification of seizure phases and types,” *Pattern Recognit*, vol. 121, 2022, doi: 10.1016/j.patcog.2021.108226.

[30] B. Gopal and A. Ganesan, “Real time deep learning framework to monitor social distancing using improved single shot detector based on overhead position,” *Earth Sci Inform*, vol. 15, no. 1, 2022, doi: 10.1007/s12145-021-00758-4.

[31] M. Xin and Y. Wang, “Research on image classification model based on deep convolution neural network,” *EURASIP J Image Video Process*, vol. 2019, no. 1, Dec. 2019, doi: 10.1186/s13640-019-0417-8.

[32] Y. Wang, Y. Jia, Y. Tian, and J. Xiao, “Deep reinforcement learning with the confusion-matrix-based dynamic reward function for customer credit scoring,” *Expert Syst Appl*, vol. 200, 2022, doi: 10.1016/j.eswa.2022.117013.