Analyzing Classroom Behavior to Enhance Student Engagement and Learning Outcomes

Mrs.Nirmala Anandhi Y

*Professor of Artificial Intelligence and Data Science*

*Rajalakshmi Engineering College* Chennai, India [email]@rajalakshmi.edu.in

Benjamin Nicolas S

Artificial Intelligence and Data Science Rajalakshmi Engineering College Chennai, India [221801005@rajalakshmi.edu.in](mailto:221801014@rajalakshmi.edu.in)

# Charless Binny K

*Artificial Intelligence and Data Science Rajalakshmi Engineering College* Chennai, India [221801007@rajalakshmi.edu.in](mailto:221801017@rajalakshmi.edu.in)

## *Abstract*— Patients with diabetes, who are much more vulnerable to cardiovascular complications because of their metabolic condition, are at serious danger from cardiovascular illnesses, especially myocardial infarction (MI), which are the world's leading causes of morbidity and mortality. Improving treatment results, reducing long-term health effects, and perhaps saving lives depend on early myocardial infarction identification. Conventional diagnostic techniques, such manually interpreting electrocardiograms (ECGs), are popular but have drawbacks. These include the process's time-consuming nature, reliance on expert interpretation, and vulnerability to human mistake. With an emphasis on patients with diabetes and those with myocardial infarction, we will compile an extensive collection of ECG images. To improve quality and standardise formats, preprocessing will be applied to the photos. To identify the existence of myocardial infarction and classify the ECG images, a convolutional neural network (CNN) will be used to extract pertinent information. To guarantee accuracy and resilience, the model will be trained and validated using methods like cross-validation.

## Metrics including accuracy, precision, recall, F1 score, and AUC-ROC will be used to evaluate the constructed model's performance. Our goal is to improve patient outcomes and enable prompt interventions by offering a dependable diagnostic tool that can help medical practitioners detect myocardial infarction in diabetes patients. This experiment demonstrates how machine learning may improve clinical decision-making and revolutionise cardiovascular diagnosis.

***Keywords— Myocardial Infraction Detection, Electrocardiogrgram(ECG) Analysis,Diabetic Patients, Convolutional neural network (CNN), Machine Learning in Healthcare.***

1. INTRODUCTION

Cardiovascular illnesses, notably myocardial infarction (MI), are primary causes of death globally. Because diabetes affects the heart and blood arteries, diabetic patients are especially susceptible to cardiovascular problems among high-risk groups. In order to improve survival chances and avoid serious health implications, myocardial infarction must be detected early. Medical practitioners manually interpret electrocardiogram (ECG) signals in order to diagnose MI. This process can be laborious and prone to human error. This strategy also demands experience and may lack consistency between multiple interpreters. Furthermore, it is challenging to precisely identify myocardial infarction due to minute alterations in ECG patterns, particularly in diabetic patients.

This project focusses on developing a machine learning model, leveraging deep learning techniques, to detect myocardial infarction early by analysing ECG images. The model uses a convolutional neural network (CNN) to automatically extract important features from ECG images, allowing it to classify the presence or absence of myocardial infarction with greater precision. By automating the process, the project aims to provide a scalable, reliable diagnostic tool that can improve outcomes for patients, especially diabetics, by lowering reliance on manual interpretation and enabling timely interventions.

The suggested solution seeks to improve accuracy and consistency while cutting down on the amount of time needed for diagnosis by automating ECG analysis. This is particularly critical in emergency situations where quick decisions are essential. Additionally, by concentrating on diabetes patients, the model is customised to address the unique difficulties that this high-risk population faces. In order to improve patient outcomes through prompt medical intervention and lessen the strain on healthcare systems, the project's ultimate goal is to create a scalable, effective, and dependable tool for early MI detection. Future studies on the management of cardiovascular illness and machine learning methods for ECG analysis may benefit from the project's findings.

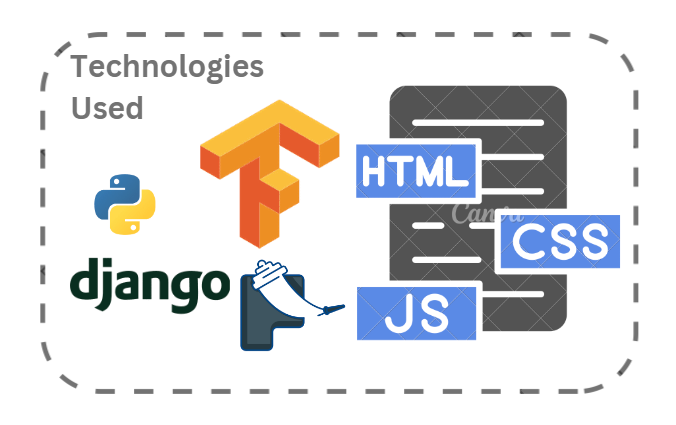


Figure 1: Tech Stacks

1. RELATED WORKS

Using electrocardiogram (ECG) data, a number of research have investigated the use of machine learning and deep learning approaches in the automated identification of cardiovascular disorders, specifically myocardial infarction (MI). Medical practitioners have historically relied on human interpretation of ECG waveforms, concentrating on important components including the P wave, QRS complex, and T wave, to diagnose myocardial infarction. These characteristics are examined for anomalies that might point to cardiac disorders.

But manual interpretation takes a lot of time, is prone to human error, and frequently calls for specific knowledge. Accurate and prompt diagnosis is made more difficult by the intricacy of ECG signals, which can differ greatly between people and especially in high-risk populations like diabetic patients. Given these drawbacks, machine learning models have become useful instruments for automating and improving the accuracy of ECG analysis.

Traditional machine learning techniques including support vector machines (SVM), decision trees, and k-nearest neighbours (k-NN) have been extensively employed for ECG signal interpretation within the last ten years. The most pertinent elements from the ECG signal, such as wave amplitudes, intervals, or morphological patterns, are manually selected by subject-matter specialists and provided as input for classification in these models, which often rely on manual feature extraction. Although the detection of cardiovascular illnesses has shown promise with these classic models, the manual feature extraction procedure is time-consuming and frequently biassed. Furthermore, manually chosen characteristics might not fully capture the intricacy of ECG signals, particularly in diabetic patients where it might be more difficult to pick up on subtleties in the patterns. Because of this, conventional machine learning techniques might not be able to provide the degree of precision needed for trustworthy clinical use, especially for detecting MI in complex and varied populations.

The interpretation of ECGs and other medical images has been transformed in recent years by deep learning techniques, especially convolutional neural networks (CNNs). In contrast to conventional models, CNNs can automatically learn and extract pertinent features straight from images or raw ECG data, greatly reducing the requirement for human interaction. The comprehensive analysis of ECG data by these deep learning algorithms might reveal intricate correlations and patterns that conventional machine learning models or human observers might overlook. CNN-based methods are preferable at detecting myocardial infarction, as numerous studies have shown.

For instance, CNNs have demonstrated remarkable accuracy in analysing 2D representations of ECG data, frequently surpassing conventional techniques. CNNs are especially useful for real-time applications in clinical settings since they enhance the overall effectiveness of the diagnosis process and the detection of small anomalies by directly learning feature hierarchies from the data.

.

Using pre-trained models, like VGG16, ResNet, or InceptionV3, that were first trained on large general-purpose image datasets and then fine-tuned for specific tasks like ECG classification has further advanced the field of automated ECG analysis. This approach significantly reduces the time required to train models and often results in higher accuracy because these models can transfer their knowledge of feature extraction from natural images to medical images like ECGs.

Transfer learning has been successfully used in a number of studies to diagnose myocardial infarctions, with notable gains in generalisability and accuracy. Transfer learning speeds up training and enhances the model's performance in specialised tasks like heart disease detection by enabling models to leverage large-scale pre-trained networks that have already learnt to recognise important visual patterns.

There is growing recognition of the need to create specialised models for high-risk groups, like diabetic patients, even though the majority of the research in this field has concentrated on identifying cardiovascular disorders in the general population. Due to the more intricate and delicate patterns in their ECG signals, diabetic patients frequently provide special obstacles for ECG analysis, making it more challenging to identify myocardial infarction using conventional methods.

By adding diabetic-specific datasets to their machine learning models, some recent research have started to close this gap and have seen encouraging outcomes. These studies have shown that models trained on the ECG data of diabetic patients can increase the accuracy of diagnosis, especially when it comes to identifying early myocardial infarction symptoms. To completely comprehend the distinctive features of ECG signals in diabetic populations and to create models that are suited to their particular requirements, more research is necessary.

In summary, the corpus of work on deep learning and machine learning techniques for ECG analysis shows great promise for enhancing myocardial infarction detection. With gains in accuracy and efficiency above conventional machine learning models, CNNs and transfer learning have become the most promising approaches. These developments are especially helpful in automated ECG data analysis, where prompt and accurate diagnosis is critical.

Notwithstanding these achievements, additional study is required to address the particular difficulties that diabetic individuals face. The development of more comprehensive, flexible, and efficient diagnostic methods will depend heavily on specialised models that can reliably identify myocardial infarction in this high-risk population. Better patient outcomes and more individualised treatment options could result from this ongoing study, which has the potential to revolutionise the diagnosis and management of cardiovascular illnesses.

1. PROPOSED SYSTEM

## System Overview

The architecture of the Student Behavior Analysis System is designed to provide a comprehensive and automated solution for monitoring student engagement in real-time. There are three main steps to the system:

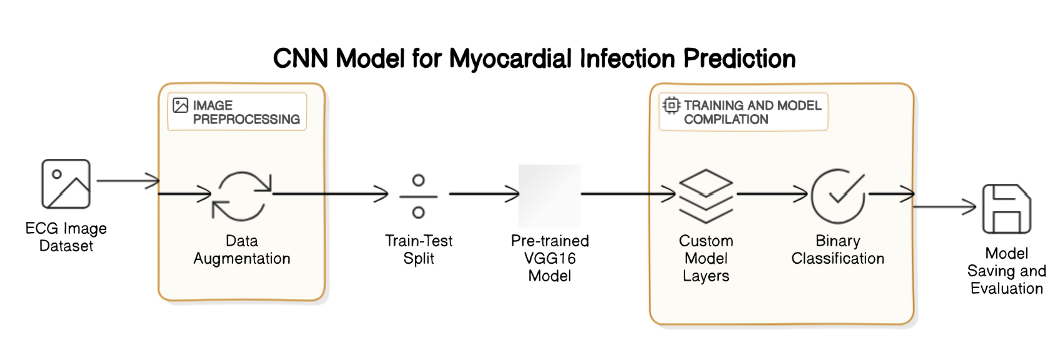


Figure 2: Overview of the System

This system is a complete pipeline that uses a Convolutional Neural Network (CNN) model specifically made to evaluate ECG images to predict myocardial infarction. An ECG image dataset with annotated images that individually reflect a normal or infarcted heart condition serves as the starting point for the procedure. A deep learning model is trained using these images to distinguish between cardiac patterns that are healthy and those that may be at risk.

Techniques for data augmentation are used to increase the dataset's variability and stop the model from overfitting. The model learns to identify significant features despite small changes in the data by creating slightly modified replicas of the original images (by flipping, rotating, scaling, or adding noise). To increase the model's resilience and make sure it works well in practical situations, this stage is essential.

The dataset is divided into Training and Testing Sets after data augmentation. The model is trained on the training set, and its performance is evaluated on the testing set. A pre-trained VGG16 model, which is renowned for its efficiency in extracting high-level features from images, forms the basis of the model architecture. Training time is reduced and complex patterns in ECG images can be captured by using this pre-trained model as a feature extractor.

The architecture is expanded by adding Custom Model Layers on top of the VGG16 backbone. By adjusting the VGG16's generalized picture classification capabilities to the subtleties of ECG data, these extra layers optimize the model for the particular task of identifying myocardial infarction patterns. The model can perform Binary Classification because to this bespoke structure, which categorizes each input image as either showing a myocardial infarction or a healthy heart.

The model is rigorously tested on the testing set after training is finished, with accuracy, precision, recall, and other metrics being measured to verify the model's dependability. Following training, the model is preserved for later analysis or clinical deployment during the Model Saving and Evaluation phase. This last phase is essential for incorporating the model into practical applications, where it can help medical practitioners promptly and precisely detect myocardial infarction symptoms from ECG images, facilitating prompt and perhaps life-saving treatments.

## System Architecture

Convolutional neural networks (CNNs), a type of deep learning, are used in the system architecture for myocardial infarction prediction utilizing ECG images in order to identify visual patterns in the data. Accurate model training and useful deployment in medical contexts are made possible by this architecture's modular components, which simplify the process from raw data to predictive insights.

An ECG image collection that has been labeled to show either normal heart activity or symptoms of myocardial infarction serves as the starting point for the procedure. These pictures serve as the model's input and are preprocessed to enhance its generalizability and performance. In order to provide a more varied training set, preprocessing entails data augmentation, which applies different transformations to the images, including rotations, flips, scaling, and random noise. This helps the model generalize better when it comes to fresh data by making it more resilient to minute changes in ECG pictures that could happen in clinical settings.

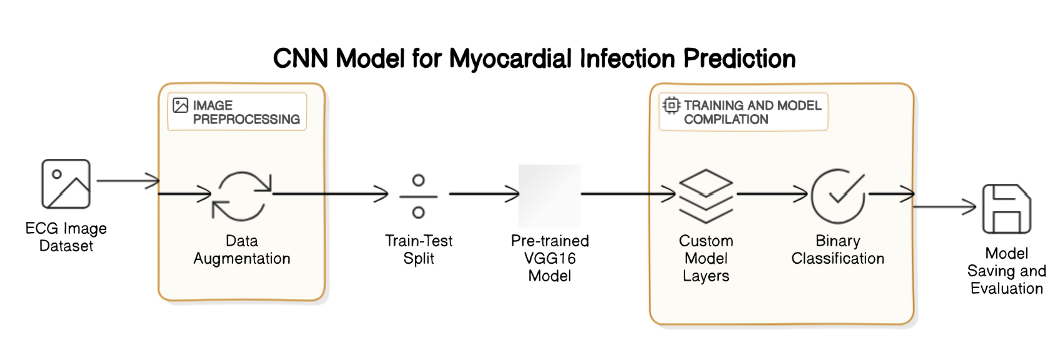


Figure 3: Architecture of the Project

The dataset is divided into training and testing subgroups following preprocessing. While the testing set assesses the model's performance, the training set is used to fit the model. Transfer Learning with a Pre-trained VGG16 Model, a deep CNN first trained on a sizable, general-purpose picture dataset (ImageNet), forms the basis of the model's fundamental design. Because of its reputation for capturing intricate spatial hierarchies in pictures, VGG16 is a perfect starting point for feature extraction from ECG pictures. The system gains from its learnt feature maps, which capture crucial visual patterns, by starting with VGG16. Because the model does not have to learn features from scratch, this method saves training time and computational resources.

The end of the VGG16 structure has Custom Model Layers added to it in order to modify it to meet the unique needs of myocardial infarction diagnosis. These layers are tuned for binary classification and customized for ECG data. In order to capture higher-order patterns unique to ECG signals, the custom layers usually consist of many dense (completely linked) layers with activation functions, such as ReLU for hidden layers. The last layer generates a binary output that indicates whether myocardial infarction is present or not using a sigmoid activation function. The architecture is able to concentrate on the subtle variations that differentiate myocardial infarction from regular heartbeats because of the unique layers.

During the **Training and Model Compilation** stage, the model undergoes training on the augmented and preprocessed ECG images, with backpropagation updating the weights to minimize the error in classification. The training process involves a careful balance of hyperparameters, such as learning rate, batch size, and the number of epochs, to ensure that the model converges effectively. To further enhance the model's predictive accuracy, cross-entropy loss is used as the loss function, optimizing the weights to differentiate between the two classes. Additionally, dropout layers may be included in the architecture to prevent overfitting, especially given the relatively limited size of ECG datasets compared to general image datasets

Following training, the model's performance is assessed on the test set, where its predictive capacity is gauged using a variety of measures, including accuracy, precision, recall, and F1 score. The model is saved and ready for use in clinical or research contexts if it satisfies the necessary performance requirements.

Lastly, the model can be saved for later use thanks to Model Saving and Evaluation. In order to forecast possible myocardial infarctions, the saved model can be imported and incorporated into medical applications. In these applications, it functions as a decision-support tool by evaluating incoming ECG images. This architecture makes it easier to classify data quickly and accurately, which makes it a useful tool for clinical practice's early diagnosis and intervention. As fresh data becomes available, the system's modular design enables upgrades and enhancements, making it flexible for upcoming advancements in ECG-based myocardial infarction identification.

## System Workflow

The system workflow for myocardial infarction prediction using ECG images is a step-by-step procedure that generates actionable predictions by processing raw ECG image data. By handling data preparation, model training, and prediction, this workflow turns raw ECG images into a trustworthy classification output that shows whether myocardial infarction is present or not.

The procedure starts with the capture and gathering of ECG images from a variety of sources, such as image archives or clinical databases. To help the model learn to differentiate between them during training, these photos are structured and labeled to reflect their respective classes (e.g., "normal" or "myocardial infarction"). Each ECG image is subsequently transformed through a number of steps to improve its quality and diversity as part of the Data Preprocessing and Augmentation process. The rotation, scaling, and flipping adjustments used in this augmentation method help to expand the dataset's effective size and expose the model to a variety of visual patterns that might be present in actual ECG readings.

The dataset is divided into training and testing subsets after the photos have undergone preprocessing. This division guarantees that the model is trained on a subset of the data and assessed on an additional, hidden subset to determine how well it generalizes. The VGG16 Pre-trained Model serves as the first feature extractor once the training data is given into the model. In order to extract general features that capture crucial spatial and textural characteristics, VGG16 processes the images through a number of convolutional and pooling layers. The VGG16 architecture's Custom Model Layers receive these feature representations and refine the feature maps especially for ECG classification. These unique layers are intended to help distinguish between patterns suggestive of myocardial infarction and regular cardiac rhythms.

Backpropagation, which iteratively modifies the model's weights to reduce classification errors, is used to optimize the model during the Model Training and Compilation phase. After every data run through the model, the loss (or error) is calculated, and the model parameters are changed to minimize this loss. With the help of the selected hyperparameters, such as learning rate and batch size, gradient descent is used to do this. Here, dropout regularization is frequently employed to avoid overfitting and make sure the model is resilient to fresh data. In order to detect indications of overfitting or underfitting and make necessary adjustments, the model's performance is routinely assessed on a validation set.

The model is finally evaluated on the testing set when training is finished. In order to determine its actual predictive accuracy, it makes predictions on test photos that it did not view during training. A thorough grasp of the model's performance is provided by the computation of metrics including accuracy, precision, recall, and F1 score, with an emphasis on the model's capacity to precisely detect myocardial infarction patients. The model is deemed ready for deployment if the outcomes are satisfactory.

The trained model is saved in a format that is simple to load and utilize in clinical applications in the last stage, Model Saving and Deployment. After that, the model may be used in a real-world setting to process fresh ECG data and instantly produce categorization results to help medical professionals. The model can accept incoming ECG images in a clinical scenario, preprocess them in real-time, and generate a prediction that indicates whether the image is likely to show a myocardial infarction or a normal cardiac condition. Healthcare practitioners can then evaluate this prognosis as part of the diagnostic procedure, enabling prompt action if required.

From data collection to preprocessing, model training, assessment, and deployment, this workflow streamlines the interpretation of ECG images, making it a dependable and effective method for detecting myocardial infarction. Each step's modular design promotes flexibility, allowing the workflow to be updated or improved as new information or methods become available. This comprehensive method guarantees that the model may produce significant insights in practical clinical applications in addition to learning from historical data.

1. WORKING PRINCIPLE

## Introduction to System Workflow

The workflow of the Student Behavior Analysis System begins with a user, generally a teacher or administrator, uploading a classroom video through an interface developed with Flask, HTML, CSS, and JavaScript. The video may be uploaded and delivered to the server with ease thanks to this interface, and it is kept there in a special Video Storage Database (D1). After that, the video is viewed and goes through a Pre-Processing stage in which each frame is taken out and ready for examination. Pre-processing tasks like as resizing, normalization, and frame quality correction are carried out in order to guarantee uniformity in a variety of classroom settings. After pre-processing, the frames are sent into a Convolutional Neural Network (CNN) model that has been trained to identify certain behaviors in the classroom, like raising hands, nodding off, and sleeping. The CNN examines every frame, identifying unique patterns of activity in real time.

The behavior detection process is followed by Action Detection and Classification, where the behaviors are categorized into pre-established groups by the system. Every student's activity is recorded with the associated timestamp, offering comprehensive data on student participation in the classroom. After that, this data is kept for later examination in an Action Data Store (D2). The system creates an extensive Engagement Report, which offers insights on student behavior during the class, based on this action data. Teachers can better grasp the dynamics of the classroom as a whole with the help of the report, which provides information on involvement frequency and disengagement incidents. In addition to this report, the system produces Suggestions for Classroom Improvement to assist the instructor in modifying their methods and boosting participation as needed. The Visualization and Feedback procedure happens at the last stage. Here, the system shows a Pie Chart that illustrates how various behaviors—like arguing, sleeping, reading, laughing, texting, or doing unknown things—were distributed during the session. Additionally, the technology gives teachers textual feedback that summarizes these behavioral tendencies, enabling them to keep a closer eye on student participation. The goal of this feedback is to provide instructors with timely, data-driven insights so they can make wise judgments. The entire system's workflow converts unprocessed video footage from classrooms into a wealth of information that teachers can use to enhance classroom management and instructional strategies and raise student engagement and learning objectives.

## Algorithm

Step 1: Video Upload and Preprocessing

* Accept a user-submitted classroom video using the online interface.
* Save the video file that was uploaded to the server.
* To read and process the video frame by frame, use OpenCV.

Step 2: Preprocessing and Frame Extraction:

* Adjust the frame's dimensions to match the 224x224 pixel input size that the CNN model requires.
* Scale the pixel values to a range of 0 to 1 to normalize the pixel values.
* Transform the frame into a model-compatible format (such as RGB).

Step 3: Recognition and Categorization of Actions:

* Feed the trained CNN model with the preprocessed frame.
* Anticipating the action for the frame, the model will categorize it as one of the actions (e.g., listening, raising a hand, leaning, using a phone, etc.).
* Compile your predictions for every frame and note how frequently each action happens.

Step 4: Analysis and Postprocessing:

* Aggregate the predictions across all frames to estimate the frequency of each action.
* Over the course of the film, tally the instances of each classed activity.

Step 5: Generate Feedback and Suggestions:

* Determine the positive behaviors (such raising your hands) based on the action numbers.
* Determine the detrimental behaviors (such as leaning and phone use).
* Based on the observed behaviors, make recommendations to the teacher on how to increase student engagement (e.g., encourage more hand-raising, limit distractions).

Step 6: Illustration:

* To see how the activities are distributed, create a pie chart.
* A detected action during the video is represented by each section of the pie chart.
* Put the analysis results and the pie chart on the webpage.

Step 7: Animation during Video Playback and Loading:

* Give the user the option to watch the analyzed video again.
* To demonstrate progress while the movie is being examined, use a loading animation.

Step 8: Results and Output Display:

* present the user with a pie chart, action counts, and feedback via the web interface.
* let the user download the analysis report.

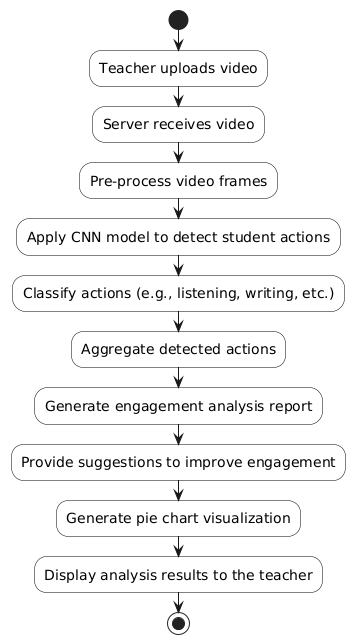


Figure 8: Algorithm of System

1. RESULT AND CONCLUSION

**Result**

The classroom footage is uploaded, processed by the system, which then produces a thorough analysis that highlights behaviors like raising hands, nodding off, and sleeping. A pie chart with the findings shows how the student actions that were seen in class are broken down clearly. It might indicate, for instance, that 25% of the time students raised their hands, 15% were not paying attention, and 50% were actively engaged in the lesson. In addition to the pie chart, the system provides comprehensive feedback that recommends ways to enhance instructional approaches by taking into account observed behaviors. Instructors can obtain a comprehensive report that includes behavior breakdowns and suggestions, enabling them to make data-driven modifications to improve student participation in the classroom.

## Conclusion

In conclusion, this project presents a significant advancement in the realm of classroom behavior analysis through the application of an optimized Fast R-CNN model. By effectively detecting critical student behaviors such as hand-raising and sleeping from video recordings, our proposed system empowers educators with valuable insights into student engagement and classroom dynamics. The utilization of advanced object detection techniques allows for accurate and real-time analysis, overcoming traditional limitations associated with manual observation methods. As a result, teachers can make informed decisions to enhance their instructional strategies and foster a more engaging learning environment. Furthermore, this system lays the groundwork for future research in automated classroom analysis, with potential applications extending beyond behavior detection to include broader aspects of student interaction and performance. Ultimately, this innovative approach not only benefits educators but also contributes to the overall improvement of educational outcomes, ensuring that every student has the opportunity to succeed.

REFERENCES AND RESOURCES

1. Karpathy, A., & Fei-Fei, L. (2015). “Deep Visual-Semantic Alignments for Generating Image Descriptions,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4), 664-676.
2. Nguyen, T. D., & Kiyomoto, S. (2018). “Analyzing Student Behavior in Classrooms Using Convolutional Neural Networks,” *Journal of Educational Technology & Society*, 21(2), 4-17.
3. Liu, W., et al. (2016). “SSD: Single Shot MultiBox Detector,” *European Conference on Computer Vision (ECCV)*.
4. Donahue, J., et al. (2014). “Decaf: A Deep Convolutional Activation Feature for Generic Visual Recognition,” *Proceedings of the 31st International Conference on Machine Learning (ICML)*.
5. Khan, A. F., et al. (2020). “A Comprehensive Review on Student Engagement Detection and Classification in Video Lectures: Current Trends and Future Directions,” *Journal of Educational Computing Research*, 58(5), 1-30.
6. Li, Y., & Chen, Z. (2017). “Deep Learning-Based Object Detection for Visual Classroom Analysis,” *Journal of Educational Technology & Society*, 20(4), 83-93.
7. Chen, T., et al. (2018). “Video-based Student Engagement Detection: A Review of the State-of-the-Art Techniques and Future Directions,” *Education and Information Technologies*, 23(5), 2187-2208.
8. Xu, H., & Zhang, S. (2019). “Real-Time Video-Based Student Engagement Detection Using LSTM,” *IEEE Transactions on Learning Technologies*, 12(2), 195-206.
9. Abdelhamed, A., & Kassem, R. (2020). “A Framework for Classroom Engagement Detection Using Neural Networks,” *International Journal of Interactive Mobile Technologies*, 14(6), 150-160.
10. Bahl, S., & Gupta, V. (2020). “Analyzing Student Attention in Lecture Videos Using Deep Learning,” *International Journal of Information and Education Technology*, 10(5), 415-421.
11. Cao, Y., et al. (2019). “Deep Learning for Classroom Behavior Analysis: A Survey,” *Journal of Educational Technology & Society*, 22(3), 119-134.
12. Ma, L., et al. (2021). “Multi-Task Learning for Student Engagement Detection in Online Learning Environments,” *Computers & Education*, 168, 104196.
13. Kaur, A., Mustafa, A., Mehta, L., & Dhall, A. (2018). “Prediction and Localization of Student Engagement in the Wild,” *2018 Digital Image Computing: Techniques and Applications (DICTA)*, IEEE, 1-8.
14. Thomas, C., & Jayagopi, D.B. (2017). “Predicting Student Engagement in Classrooms Using Facial Behavioral Cues,” *Proceedings of the 1st ACM SIGCHI International Workshop on Multimodal Interaction for Education*, 33-40.
15. Whitehill, J., Serpell, Z., Lin, Y.C., Foster, A., & Movellan, J.R. (2014). “The Faces of Engagement: Automatic Recognition of Student Engagement from Facial Expressions,” *IEEE Transactions on Affective Computing*, 5(1), 86-98.