EduGaze: Gaze Based Attention Tracking System

Anju Prabha[1], Shariba Kouzer[2], Sneha Denny[3], Mahima Muraleedharan[4], Blessy T Sunny[5]

[1]*Asst Professor, Dept of AI & CY, Sree Narayana Gurukulam College of Engineering, Kochi, India*
anjuprabha@sngce.ac.in

[2]*Student, Dept of CSE, Sree Narayana Gurukulam College of Engineering, Kochi, India*
sharibakouzer@gmail.com

[3]Student, Dept of CSE, Sree Narayana Gurukulam College of Engineering, Kochi, India
snehadenny995@gmail.com

[4]*Student, Dept of CSE, Sree Narayana Gurukulam College of Engineering, Kochi, India*
mahimamuraleedharan.xib@gmail.com

[5]*Student, Dept of CSE, Sree Narayana Gurukulam College of Engineering, Kochi, India*
blessytsunny@gmail.com

---***---

Abstract - This paper introduces EduGaze, a web-based system designed to monitor and analyze student engagement in real time during online classes. The platform is built using a scalable monorepo architecture that combines a Node.js/Express backend, a React-based dashboard for teachers, and a Chrome extension for collecting data from students. It uses Socket.io to enable a low-latency, two-way communication stream, allowing the system to continuously track attention states such as focused, distracted, and offline. These interactions are synchronized and stored in an optimized database, supporting both real-time visualization and later analysis. The system also includes an AI module powered by a locally hosted large language model, which generates automated session reports, identifies attention patterns, and suggests ways to improve student focus. Experimental results indicate that EduGaze can effectively detect engagement trends and provide meaningful insights to educators. By eliminating the need for specialized hardware and cloud-based AI services, the system offers a cost-effective, privacy-friendly, and scalable solution for enhancing online learning experiences.

Key Words: Eye gaze tracking, real-time analytics, student engagement, human-computer interaction, machine learning

1.INTRODUCTION

In the shift towards digital and hybrid learning environments, maintaining student engagement has become a major challenge for educators. Traditional cues such as eye contact and body language are absent in virtual classrooms, making it difficult for teachers to assess whether students are attentive or distracted. This creates a visibility gap that affects both teaching effectiveness and student learning outcomes.

To address this issue, the proposed system EduGaze introduces a browser-based solution that uses webcam-based gaze tracking and machine learning to monitor student attention in real time. The system analyzes gaze direction and facial cues to determine whether a student is focused on the screen or distracted. Unlike traditional monitoring tools, EduGaze is designed as a supportive system that enhances learning rather than enforcing strict surveillance.

EduGaze provides real-time notifications to teachers when students remain inattentive beyond a certain threshold, enabling timely intervention. It also generates automated session summaries that highlight individual and class-level engagement. With the integration of a Node.js and Socket.io backend, the system supports real-time communication, AI-generated reports using Ollama, and features like content recovery assistance and seamless sharing of study materials.

The system follows a privacy-first approach, where all gaze analysis is performed locally within the browser, eliminating the need for specialized hardware and preventing external data storage. By combining computer vision, artificial intelligence, and web technologies, EduGaze offers a reliable, cost-effective, and ethical solution to improve engagement, accountability, and overall learning quality in online education.

2. RELATED WORKS

Eye gaze tracking and student engagement monitoring have been widely studied in the fields of human-computer interaction and educational technology. Earlier methods mainly depended on specialized hardware-based eye trackers, such as those developed by Tobii, which offer high accuracy and reliability. However, these systems are expensive and require dedicated equipment, making them impractical for large-scale or low-cost use in educational settings.

With recent technological progress, attention has shifted toward webcam-based eye tracking solutions. Tools like WebGazer.js allow gaze estimation using standard cameras and web browsers. These systems rely on machine learning models along with user calibration to map eye movements to screen positions. While they remove the need for specialized hardware, their performance can be affected by factors such as lighting conditions, camera quality, and user movement.

At the same time, modern learning management systems like Moodle and Google Classroom have incorporated analytics features that track student activity, assignment submissions, and participation. Despite these capabilities, they do not provide real-time attention tracking or detailed insights into student engagement during live sessions.

More recently, advancements in artificial intelligence have led to the development of AI-driven educational tools powered by models such as GPT models. These systems can generate personalized feedback and summarize learning content, but they often depend on cloud-based services, raising concerns about latency, operational costs, and data privacy.

The proposed EduGaze system addresses these limitations by combining real-time engagement monitoring, browser-based gaze tracking, and local AI-powered analytics into a single integrated platform. Unlike existing approaches, it focuses on low-latency communication, privacy-friendly local processing, and seamless compatibility with online learning environments. This makes it a scalable, cost-effective, and intelligent solution for monitoring and improving student engagement.

3. PROPOSED SYSTEM

EduGaze is a browser-based system developed to monitor student attention during online classes. It uses a webcam to capture eye movement and applies machine learning techniques to understand whether a student is focused or distracted. This helps teachers improve engagement and take timely actions during live sessions.

3.1 System Overview

The system is built using several key components that work together:

- A Chrome extension that runs on the student's browser
- Webcam access through the getUserMedia API to capture video input
- Gaze tracking implemented using WebGazer.js
- Machine learning models such as Ridge Regression and a Neural Classifier for attention prediction
- A backend server developed with Node.js and Socket.io for communication
- A teacher dashboard to display real-time student attention data

The system continuously analyzes facial and eye movements to estimate where the student is looking. Based on this data, it determines the level of attention and updates the teacher dashboard instantly, allowing teachers to monitor the class effectively.

3.2 Working and Features

- Tracks eye movement in real time to detect attention levels.
- Categorizes students as attentive, distracted, or inactive
- Sends instant alerts to teachers when a student loses focus
- Provides an AI-based assistant to help students regain attention
- Generates session-wise reports showing engagement patterns
- Allows easy sharing of notes and study materials

This system provides a more interactive and responsive learning environment. By combining real-time tracking with AI-driven insights and reports, EduGaze supports both teachers and students in improving the overall online learning experience.

4. SYSTEM ARCHITECTURE

EduGaze is built as a high-performance Monorepo using the pnpm workspace manager. This structure ensures atomic updates across all system components and promotes code reuse between the frontend and backend modules.

4.1 Core Modules

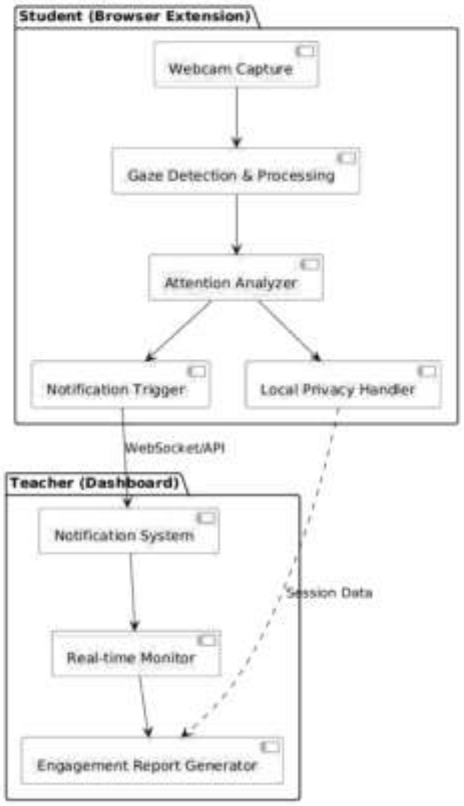

Fig -1: System Architecture

- Backend API (Node.js/Express): The central nervous system that manages session persistence, real-time event routing via Socket.io, and interfaces with the local AI engine.
- Teacher Dashboard (React/Mantine): A sophisticated single-page application (SPA) that transforms raw event data into actionable visualizations, student timelines, and session reports.
- Student Extension (Chrome/Manifest V3): A lightweight client-side collector that monitors engagement metrics in Google Meet and provides the EduGazer AI assistant to students.

4.2 System Workflow

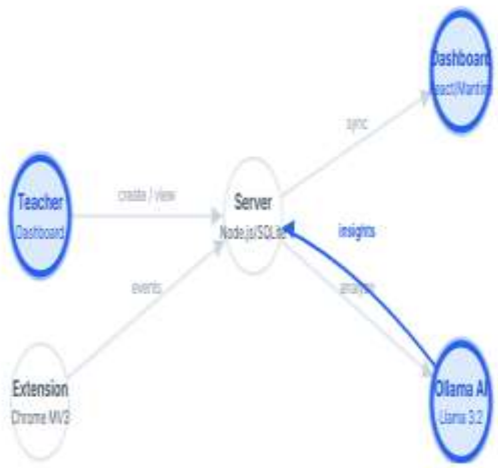

Fig -2: System Workflow

The workflow of the system begins with the initialization of the application, where the user is prompted to allow access to the webcam. Once permission is granted, the system starts capturing video frames continuously and processes them directly within the browser in real time. Using computer vision techniques, it identifies the user's face and extracts the eye regions from each frame, converting these visual details into numerical features for further processing.

After extracting these features, the system moves into a calibration phase. During this stage, user interactions such as mouse clicks or cursor movements are recorded along with the corresponding eye data. These calibration points are used to train a regression model that learns to map eye characteristics to specific screen coordinates. As the user continues interacting, the model keeps updating itself, gradually improving its prediction accuracy through continuous self-calibration.

Once the calibration is complete, the system begins real-time gaze prediction. The extracted eye features are passed into the trained machine learning model, which estimates the user's point of focus on the screen in terms of X and Y coordinates. To make these predictions more stable and reliable, smoothing techniques are applied to reduce noise. The final gaze positions can then be visualized or used within applications, such as tracking a student's attention on presentation slides.

This entire process runs in a continuous loop, where each video frame is captured, processed, and converted into gaze predictions instantly. Because all computations take place within the browser itself, the system achieves low latency while also maintaining user privacy, as no video data needs to be sent to external servers.

5. RESULTS AND DISCUSSION

EduGaze works directly within the browser and uses the webcam to observe how a student behaves during an online class. Instead of relying on a single parameter, it considers a combination of eye movement, blinking patterns, and head position. Bringing these together helps the system estimate whether a student is attentive or distracted. Since everything happens in real time, it can be used alongside platforms like Google Meet without requiring screen sharing, making it convenient and less intrusive.

When tested in a controlled environment, the model achieved an accuracy of around 88.6%, with a loss value close to 0.41. These results indicate that the training process was fairly stable and that the model was able to learn meaningful patterns from the data. It performed well in distinguishing between focused and distracted behaviour under these conditions. The AUC score of 0.93 further supports the model's reliability. In addition to this, a feature called EduGazer AI was introduced, which generates short summaries and helps students catch up on missed content. This proved especially helpful during longer sessions where maintaining continuous attention can be difficult.

5.2 Testing Performance

The same system was then evaluated in actual classroom scenarios. Unlike the controlled setup, this involved variations in lighting, device quality, and natural student behaviour. Under these conditions, the accuracy dropped to about 76.8%.

This decrease is understandable because real-world environments are much less predictable. Factors like poor lighting, differences in camera resolution, and natural movements such as briefly looking away or adjusting posture can affect the system's observations. As a result, identifying attention becomes more challenging outside a controlled setting.

To improve stability, a short delay mechanism was introduced before triggering alerts. The system waits for about 5 to 10 seconds of continuous distraction before issuing a warning. This helps reduce false alerts caused by momentary movements. During testing, this approach made the system noticeably more reliable for continuous use. Additionally, teachers were able to share notes and materials during sessions without interfering with the tracking process.

5.3 Behavioural Analysis

The system performs well when behaviour patterns are clear. For example, if a student consistently looks at the screen, it is accurately identified as attentive. Similarly, prolonged distraction is also detected effectively.

However, minor and brief actions still pose some challenges. Quick glances away from the screen or slight head movements are sometimes misclassified as distraction. These are normal behaviours during any class, but they can disrupt the pattern the model relies on. Because of this, the system may occasionally produce incorrect classifications. This is an area that can be refined in future improvements.

Despite these limitations, the system still provides a useful overview of student engagement. The generated reports allow teachers to observe general behaviour trends, which can help them adjust their teaching strategies if needed.

5.4 End to End Validation

The entire system was tested as a complete working pipeline within the browser. It begins with webcam input, followed by face and gaze detection, and finally the model determines the student's attention level. The results are displayed on a teacher dashboard in real time.

If a student remains distracted beyond the defined threshold, an alert is triggered. At the same time, the overall class attention level is continuously updated. After the session, a detailed report is generated, including attention scores, behavioural patterns, and how engagement changes over time.

The dashboard is developed using React and Mantine, providing both live updates and access to stored reports. On the backend, Node.js with Express and Socket.io handles communication. Ollama is used for generating summaries, while SQLite stores session data. A Chrome extension is also used to capture attention metrics and integrate the EduGazer AI assistant, although further improvements are needed for smoother interaction with meeting platforms.

Since everything runs within the browser, no additional hardware is required. This keeps the system simple to use while also supporting user privacy. Overall, EduGaze shows potential as a practical tool for understanding student behaviour and improving the management of online classes.

6. LIMITATIONS

- The accuracy of the system can vary depending on the quality of the webcam and the lighting in the environment.
- External factors such as background settings and how the student positions their face may influence detection results.
- Frequent head movements or obstructions (like hands or objects blocking the face) can sometimes cause incorrect attention analysis.
- Features like live alerts and the teacher dashboard depend on a reliable internet connection to work smoothly.

7. FUTURE SCOPE

EduGaze provides a strong foundation for monitoring student attention in online classes, but there are several areas where it can be improved. One important direction is the use of advanced deep learning models like CNNs and RNNs to improve gaze detection accuracy and enable emotion recognition. This would help the system understand not just attention, but also student engagement levels such as interest or confusion.

Integration with Learning Management Systems like Google Classroom, Moodle, and Microsoft Teams can make the system more practical for real-world use. Cloud-based analytics can also be added to support long-term performance tracking and allow teachers to monitor multiple students through a single dashboard.

Additional features such as gesture recognition, speech analysis, and AI-based feedback can provide deeper insights into student participation. Future work can also focus on optimizing the system for mobile devices and low-bandwidth environments. At the same time, improving privacy controls will be important to ensure safe and ethical usage.

8. CONCLUSIONS

EduGaze improves the quality of online learning by helping teachers monitor student attention in real time through a browser-based system. It uses technologies like WebGazer.js, MediaPipe, and machine learning to provide a cost-effective solution without requiring extra hardware. Since most of the processing happens within the browser, user privacy is also maintained.

The system tracks gaze direction, head movement, and eye focus to determine whether a student is attentive or distracted. It provides real-time alerts and generates session reports, helping teachers better understand class engagement and respond quickly when needed.

EduGaze works smoothly with platforms like Google Meet and Zoom and is designed to run efficiently on regular devices. Overall, it helps bridge the gap between traditional and online classrooms by offering better visibility into student behaviour and engagement.

REFERENCES

[1] A. Papoutsaki, N. Daskalova, P. Sangkloy, J. Huang, J. Laskey, and J. Hays, “WebGazer: Scalable Webcam Eye Tracking Using User Interactions,” in Proc. of the 25th International Joint Conference on Artificial Intelligence (IJCAI), 2016, pp. 1550–1556.

[2] M. K. Hossen and M. S. Uddin, “Attention Monitoring of Students During Online Classes Using XGBoost Classifier,” International Journal of Emerging Technology and Advanced Engineering, vol. 13, no. 3, pp. 54–61, 2023.

[3] V. Vidhya and D. R. Faria, “Real-Time Gaze Estimation Using Webcam-Based CNN Models for Human–Computer Interaction,” Computers, vol. 14, no. 57, pp. 1–13, 2025.

[4] T. Robal, Y. Zhao, C. Lofi and C. Hauff, “Webcam-based Attention Tracking in Online Learning: A Feasibility Study,” in Proc. 23rd International Conference on Intelligent User Interfaces (IUI’18), Tokyo, Japan, Mar. 2018, pp. 189-197.

[5] N. Dilini, A. Senaratne, T. Yasarathna, N. Warnajith and L. Seneviratne, “Cheating Detection in Browser-based Online Exams through Eye Gaze Tracking,” 2021 6th International Conference on Information Technology Research (ICITR), pp. 1-8, Dec. 2021

[6] R. K. Rahul, S. Shanthakumar, P. Vykunth and K. Sairamnath, “Real-time Attention Span Tracking in Online Education,” International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 11, no. 9, pp. 11-17, Aug. 2022

[7] S. Wei, D. Bloemers, and A. Rovira, “A Preliminary Study of the Eye Tracker in the Meta Quest Pro,” in Proceedings of the ACM International Conference on Interactive Media Experiences (IMX ’23), Nantes, France, June 12–15, 2023, pp. 216–221.

[8] J. S. Skovsgaard, M. V. Pedersen, S. B. Risi, and I. T. Skovgaard, "Evaluation of Deep-Learning-Based Webcam Eye-Tracking Methods in Online Experiments," Behavior Research Methods, Springer, vol. 56, pp. 415–430, 2024.

[9] L. Li, Z. Liu, and C. Lin, "A Smart Eye Tracking System for Virtual Reality," International Journal of Innovative Science and Research Technology (IJISRT), vol. 9, no. 10, pp. 1023–1029, Oct. 2024.

[10] L. Falch and K. S. Lohan, "Webcam-Based Gaze Estimation for Computer Screen Interaction," *Frontiers in Robotics and AI*, vol. 11, Art. 1369566, 2024.