iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** IV **Month of publication:** April 2025

DOI: https://doi.org/10.22214/ijraset.2025.68816

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IV Apr 2025- Available at www.ijraset.com

Math-Lab: A Math Companion

Dr. Renuka Deshpande[1], Ashish Santosh Kolekar[2], Niranjan Bhatu Patil[3], Nusrat Fatima Shamsher Malik[4]
[1]*Associate Professor, Shivajirao S. Jondhale College of Engineering*
[2, 3, 4]*Student, Shivajirao S. Jondhale College of Engineering*

Abstract: ***MathLab is a substitute knowledge curriculum vitae designed to help students from elementary to high school with their studies. It offers a variety of interactive and AI-powered features to help students better understand and solve complex math problems. Key features include AI Tutor, where students can learn or review topics and even upload news to enhance learning; Air Solve, where students can draw math problems in the air and the system will instantly answer; Quick Solve, an AI-based video generator that creates videos that solve given problems and explain the solution, allowing students to extract rules and procedures to make them more understandable; and Quick Solve, which is an AI-powered video generator that allows students to extract rules and procedures to make them more understandable. The platform also includes AI-powered automatic code generation and repair, where the system generates rules to solve problems and self-corrects errors. The main goal of MathLab is to provide ongoing academic support to weak and struggling students by enabling self-paced and successful learning.***
Keywords: ***AI-Powered Learning, Interactive Mathematics, Gesture Recognition, Automated Problem Solving, Dynamic Graph Visualization, Code Generation Automation, Code Fixation Automation.***

I. INTRODUCTION

MathLab is an innovative AI-based platform designed to improve the learning experience for students from standard 1 to HSC who struggle with traditional teaching methods or who do not have access to personalized academic support. The goal of the project is to bridge the educational gap by providing a variety of interactive and dynamic learning features that make mathematics more accessible, fun, and understandable. MathLab's primary goal is to provide a flexible learning environment tailored to each student's needs, helping them to master mathematical concepts with ease and confidence.

In today's rapidly changing educational environment, the need for personalized learning tools has become even greater. Many students struggle to learn mathematics due to traditional teaching methods that cannot accommodate diverse learning styles or paces. MathLab addresses these challenges by introducing innovative features like AI Tutor, Air Solve, and Quick Solve that allow students to learn at their own pace with real-time assistance and interactive interaction. AI Tutor makes learning much more student-centered by providing a personalized learning path where students can choose topics, download media, and receive customized explanations.

One of MathLab's most compelling features is Air Solve, a system that uses gesture recognition technology to allow students to solve math problems simply by drawing them in the air. This feature not only makes the learning process more interactive, but also encourages students to engage with math in a non-traditional, yet fun and practical way. Air Solve tracks hand movements to instantly recognize and solve equations, turning passive learning into active and collaborative learning. Moreover, Quick Solve enhances the learning experience for students by automatically generating code based on the given math problem. This feature not only solves the problem, but also generates a video explanation that shows step-by-step how the solution was achieved. If there is an error in the code at runtime, Quick Solve will autonomously attempt to solve the problem, providing students with additional learning opportunities on how to approach and resolve coding errors. This coding automation introduces students to computational thinking and improves their problem-solving skills, and is especially helpful for students interested in mathematics and programming.

Another important feature of MathLab is the Practice Set tool, which allows students to create customized practice problems based on the topic they need help with. These practice kits are available for download in PDF format, allowing students to conveniently practice offline. This feature allows students to continuously practice, reinforce their knowledge, and track their progress, which improves their retention and acquisition of key concepts. Additionally, MathLab's graph plotting feature allows students to create dynamic graphs that allow them to visually manipulate graph parameters and see the resulting graph change immediately. This visual feedback helps students gain a deeper understanding of the relationships between mathematical functions and variables.

MathLab is an essential tool in modern education, with a focus on supporting students who are academically struggling or disadvantaged.

MathLab aims to provide personalized, accessible, and comprehensive solutions for students to succeed in mathematics by leveraging artificial intelligence and interactive learning capabilities. The platform enables students to explore mathematical concepts independently and confidently, making it an essential resource for those seeking to overcome educational barriers and enhance their learning experience. MathLab essentially combines technology and education to create a holistic and engaging learning environment that prioritizes student success.

II. LITERATURE SURVEY

[1][2][4] focus on AI for video generation and summarization. [1] explores automating educational animated videos using NLP for question-based explanations. [2] presents a neural network model for extracting video segments and generating real-time summaries. [4] uses deep learning, CNNs, RNNs, and attention mechanisms for video captioning, enhancing multimedia accessibility.

[3][5] investigate AI's creative use in video generation. [3] integrates AI-driven facial animation and speech synthesis to create dynamic talking faces. [5] uses generative AI to create digital dance visuals, promoting traditional art forms like classical dance.

[6][7] propose air-writing recognition systems using computer vision and deep learning. They focus on challenges like segmenting air-drawn characters and optimizing CNN models. Their systems perform well in controlled environments but struggle with noise and varied lighting.

[8][9] develop gesture-based air-writing detection models using OpenCV and MediaPipe for hand-tracking. They highlight real-time preprocessing techniques, such as Gaussian filtering, to improve clarity, but complex mathematical expressions remain difficult.

[10] introduces a pattern recognition-based air-writing system using a hybrid AI model (CNN + LSTM). This system improves recognition accuracy and distinguishes handwriting styles but faces real-time processing delays with cursive writing.

[11][12] explore machine learning integration in air-writing systems. RNNs perform better for continuous input, while SVMs offer faster but less accurate recognition of complex strokes.

[13] presents an AI-driven real-time numeral recognition system, focusing on handwritten digit sequences. It employs a convolutional autoencoder for noise reduction, enhancing digit recognition accuracy.

[14] introduces SCLAiR, a contrastive learning-based air-writing recognition model. It improves user independence and accuracy by recognizing diverse handwriting patterns, outperforming traditional CNN methods by 20%.

[15] explores inertial sensor-based air-writing recognition using accelerometer and gyroscope data for detecting 3D-space handwriting gestures, showing promise for wearable devices, though it faces higher error rates due to hand tremors.

[16][17] discuss AI's impact on education in India, focusing on CBSE's AI curriculum and personalized learning approaches. Challenges include teacher training, data privacy, and ethical concerns.

[18][19] analyzes AI adoption in Indian universities, highlighting AI's role in student engagement and administrative efficiency. They caution against over-reliance on AI, which may reduce critical thinking and human interaction.

[20] proposes machine learning models to enhance e-education in India, including personalized course recommendations and automated grading. The study suggests genetic algorithms to optimize course structures.

[21] examines AI's evolution from basic educational tools to intelligent tutoring systems. The research discusses AI-assisted assessments and highlights ethical concerns like algorithm bias and data security.

[22] evaluates AI adoption challenges in Indian education, such as infrastructure limitations and teacher readiness. The research emphasizes AI's potential for administrative efficiency and student engagement through gamified learning.

[23][24] improve mathematical function plotting. [23] focuses on formally verified plotting methods using the Coq proof assistant, while [24] enhances 3D scatterplots for better spatial data visualization.

[25] introduces rtopmap, an interactive research topics visualization system based on graph theory. The system allows users to explore academic trends through a 35,000-node weighted topics graph.

[26] propose a graph-based method for plotting bifurcation diagrams. This educational tool simplifies equilibrium point behaviors, offering an alternative to complex computational methods.

[27][28] explore the application of pretrained transformer models and Automated Program Repair (APR) for bug-fixing. Drain's DeepDebug improves patch generation by 33%, while Fan emphasizes the need for rigorous code quality checks in APR.

A. *Literature Gap and Overcoming*

Some of the existing approaches have endured with limitations, and our solution to such:

1) *Limitations*

- Limited Engagement: Students often find math boring or difficult due to the lack of interactivity.
- Ineffective Learning Methods for Struggling Students: Many students face difficulties in understanding complex math concepts.
- Limited Access to Academic Support: Students from underserved areas may lack access to tutors or personalized academic support.
- Inability to Visualize Mathematical Concepts: Traditional methods often fail to provide dynamic, visual feedback for concepts like graph manipulation.
- Lack of Practice Resources: Many students do not have access to enough practice material to reinforce learning.
- Slow Feedback: In traditional classrooms, students may wait for days to get feedback on assignments.
- Over-reliance on Textbook-Based Learning: Many students are restricted to textbook problems with limited real-world application.
- Lack of Automation for Problem Solving: Traditional education often lacks automation for problem-solving, making it harder for students to explore multiple solutions.
- Limited Integration of Technology for Learning: Many schools have limited access to advanced technologies for personalized learning.
- Inadequate Support for Self-Learners: Self-learners often struggle without immediate help.

2) *Overcomings*

- MathLab introduces interactive features like Air Solve and dynamic graph plotting, making learning more engaging and fun.
- MathLab provides real-time assistance through Quick Solve and AI Tutor, helping students grasp difficult concepts more effectively.
- MathLab's AI-driven system offers personalized help, reducing the gap in educational support.
- MathLab's graph plotting feature allows students to visually interact with graphs, enhancing their understanding.
- MathLab's Practice Set tool provides custom practice problems that students can download and work on offline.
- MathLab provides instant feedback, allowing students to understand mistakes immediately and correct them.
- MathLab's Quick Solve automates problem-solving and offers solutions with explanations, reinforcing the learning process.
- MathLab integrates AI and gesture recognition technology to bridge this gap, providing students with an innovative and tech-driven approach to learning.
- MathLab's on-demand features, like AI Tutor and Quick Solve, allow students to learn independently with the support they need, without the reliance on teachers or tutors.

III. METHODOLOGY

MathLab's development is based on creating a multifaceted platform that combines a variety of cutting-edge technologies to provide a rich educational experience. Each platform's core features are designed with simplicity and ease of use in mind while providing powerful functionality.

1) AI Tutor: This feature allows students to learn or review math topics interactively. Students can upload images, files, or media to receive relevant feedback, explanations, and solutions. The AI system analyzes the downloaded media and provides customized learning materials to guide students on the topics they are studying. This allows students to review specific topics according to their individual needs, making it an effective tool for self-study.
2) Air Solve: A real-time gesture recognition system built using OpenCV and MediaPipe. Students can draw equations or other math problems in the air, and the platform instantly recognizes these gestures and solves the problem. This system has the ability to recognize dynamic hand gestures and provide real-time feedback, adding a new level of interactivity to math learning and encouraging students to engage more directly with the content they are learning.
3) Quick Solve: Quick Solve is an AI-based tool that generates code to solve a given math problem. When you input a question, the system automatically generates a solution in code form, runs the code, and creates a video explaining each step of the process. If the AI detects an error while executing the code, it will autonomously attempt to solve the problem by generating new fixes and re-running the code. It will do this up to two times. This feature not only helps students understand how to approach problems, but also teaches them how to solve real-world problems using coding.

4) Practice Sets: The practice sets feature of this platform allows students to create custom practice problems based on their chosen topic. The created sets can be downloaded in PDF format, allowing students to practice and test their skills offline. This feature is especially useful for students who need additional practice to consolidate their knowledge and track their progress over time. Teachers can also assign practice sets that are tailored to the specific needs of each student.
5) Graph Plot: MathLab provides dynamic graphing capabilities that allow students to visually manipulate the values of mathematical functions and see the effects immediately. This allows students to understand the relationships between variables and how changes to one aspect of a problem affect the overall equation. This feature allows students to experiment with graphs and gain a deeper understanding of mathematical concepts that may be difficult to understand with static examples.
6) Code Generation Automation: This project includes two functions which might be dynamically generated at runtime the usage of AI. Instead of preloading static functionalities, the required additives are built on demand based totally on consumer input. The AI agent is used to generate code for exercise question technology and video animation visualization during execution, and this code is compiled and rendered via the consumer interface instantly. Dynamically at runtime. This method drastically reduces the base assignment length and minimizes CPU load, making the machine optimized for performance and appropriate for low-specification gadgets. It permits the utility to scale its capability simplest when wished, enhancing efficiency and responsiveness.
7) Code Fixation Automation: In case any errors occur in the course of the process—whether runtime, UI-related, or API-stage—the gadget captures them and stores the details in a log document based as a dictionary. These logs serve as diagnostic records for the AI agent, which analyzes the facts to become aware of the motive of the problem and provoke corrective movements routinely. This allows actual-time trouble detection and backbone, enhancing the machine's reliability and lowering the need for guide debugging.

These integrated features aim to create a comprehensive platform that supports a variety of learning needs and teaching methods, making it an ideal tool for both students and teachers.

IV. SYSTEM ARCHITECTURE

MathLab's architecture is designed to provide a seamless experience with a variety of components that work together to process input data and generate solutions. The system is modular, so each feature can operate independently, but can also be seamlessly integrated with other features.

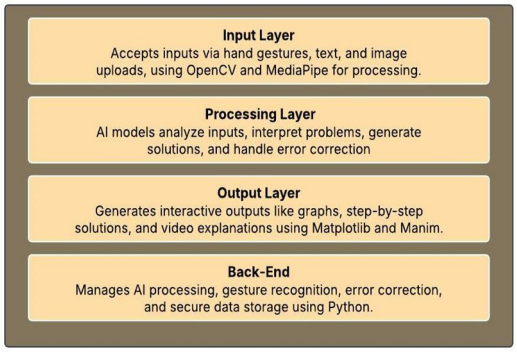


Fig 1. Architecture

1) Input Layer: The platform accepts input through a variety of channels, including hand gestures (for Air Solve), file uploads, and manual input. OpenCV is used to process video and image files, while MediaPipe facilitates gesture recognition by identifying hand landmarks, tracking motion, and solving mathematical equations drawn in the air. The platform also allows text input and image uploads to solve problems or create custom exercise sets.
2) Processing Layer: This layer is responsible for analyzing input data and generating corresponding output data. The platform's AI models use machine learning algorithms to interpret problems and generate solutions. For example, Quick Solve's AI analyzes input data, generates appropriate code, and executes it. If there is an error in the code, the system will try to correct the error by generating a new solution. In AI Tutor, the AI analyzes the uploaded file or image and provides a solution that is appropriate for the situation.

3) Output Layer: This layer generates dynamic outputs such as graphs, step-by-step solutions, and video explanations. The Matplotlib library is used to create static and dynamic graphs, and Manim is used to create high-quality animated videos that explain the solution in a more visual format. Quick Solve also generates a video of the code execution process and presents the resulting output with detailed explanations. The output is designed to be interactive, interesting, and visually informative, enhancing the learning process.

4) Backend: The backend is built using Python, ensuring efficient communication between all components. The AI model, gesture recognition, and error correction logic interact seamlessly through the backend, enabling real-time processing and dynamic output generation. Data storage of practice sets and user-generated content is handled securely to ensure user privacy and data integrity.

Together, these layers create a powerful and scalable platform that can deliver personalized and dynamic learning experiences to students across a variety of learning styles.

V. RESULTS

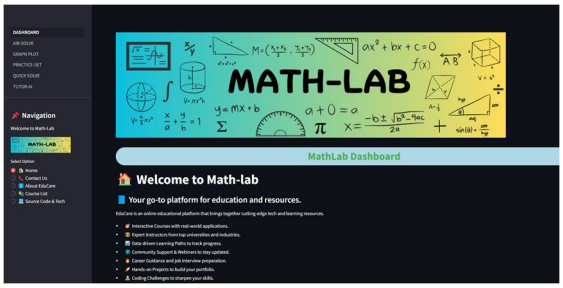


Fig 2. Dashboard

Dashboard: A user-friendly interface providing seamless navigation across multiple functionalities.

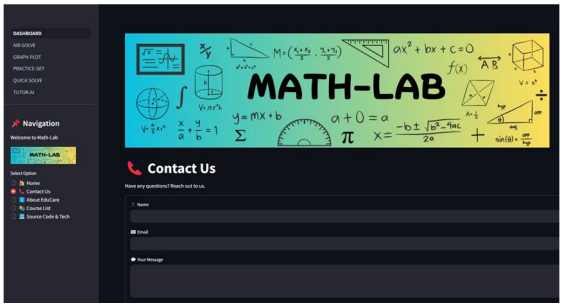


Fig 3. Contact Us

Contact Us: A support feature allowing users to seek assistance, report issues, and get queries resolved.

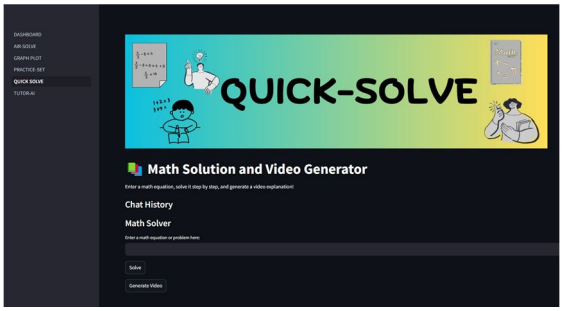


Fig 4. Quick Solve

Quick Solve: AI generates step-by-step solutions with animated visualizations for better understanding.

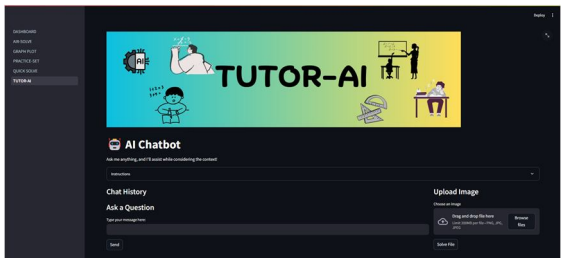


Fig 5. AI Tutor

AI Tutor: An intelligent virtual assistant offering interactive explanations, video solutions, and practice sets.

Fig 6. Practice Sets

Practice Sets: AI-powered test generator providing dynamic, personalized practice questions with explanations.

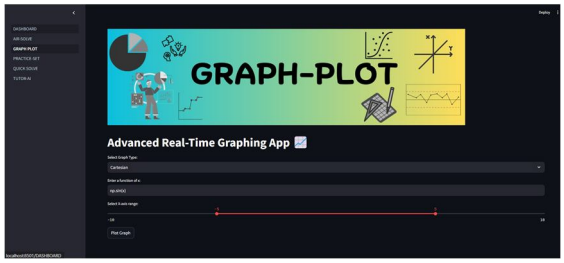


Fig 7. Graph Plot

Graph Plot: An interactive tool for creating, modifying, and analyzing graphs with real-time controls.

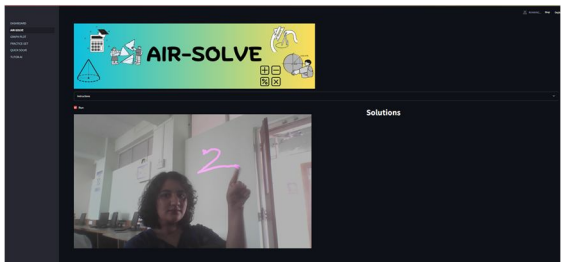


Fig 8. Air Solve

Air Solve: A gesture-based math solver using AI to recognize and solve handwritten equations instantly.

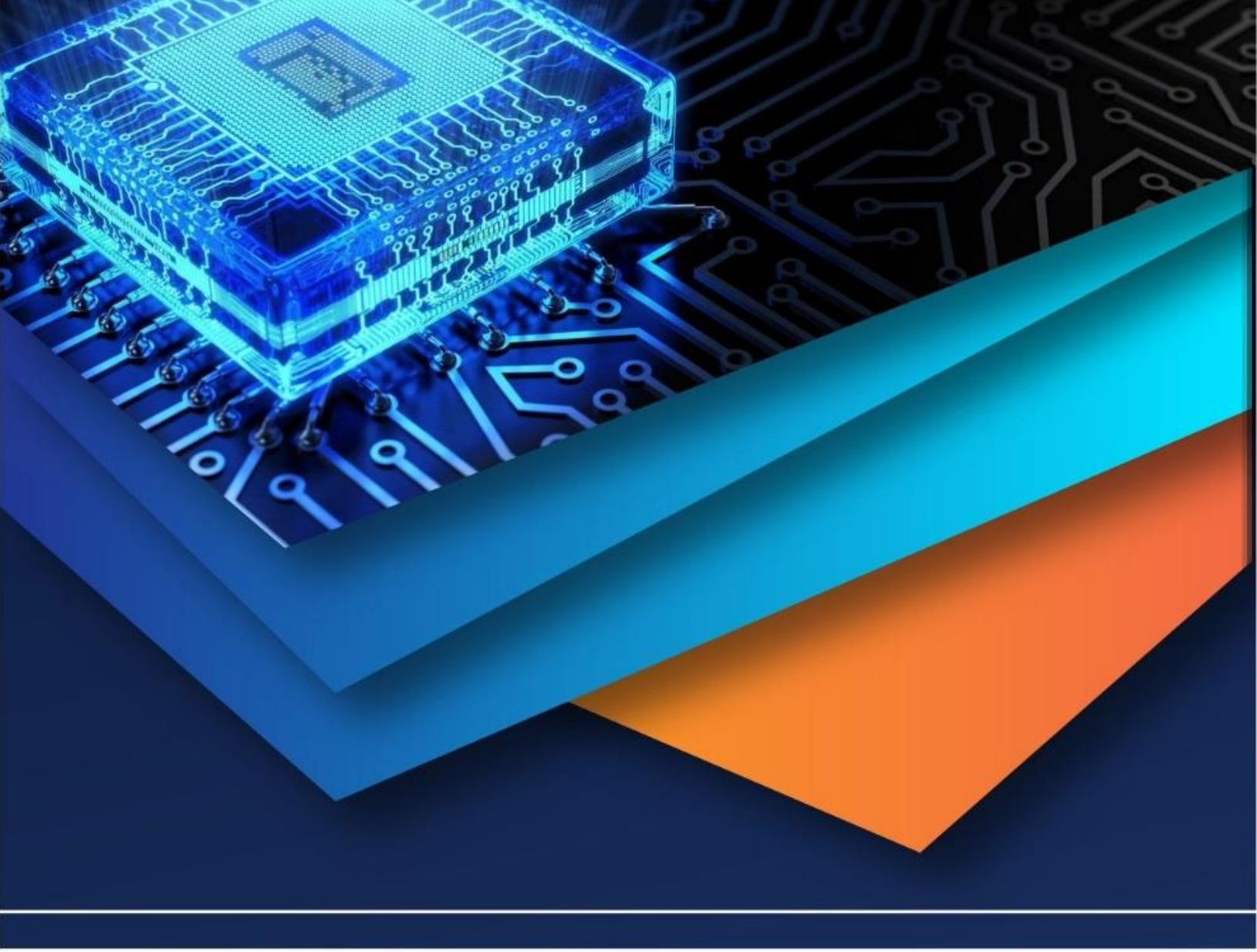
VI. CONCLUSION

MathLab is an innovative educational platform designed to provide students with advanced learning tools to help them succeed in mathematics. It uses artificial intelligence, interactive gestures, and real-time graphing to provide a fun and effective comprehensive learning experience. Features like AI tutor, quick problem solving, and air problem solving provide students with immediate help for difficult problems, while practice sets and graph plotting features provide ongoing practice and understanding. The platformâs code generation and auto-correction capabilities provide immediate solutions to coding problems, further enhancing its usability.

MathLab is not just a problem solving tool. It is a comprehensive solution aimed at facilitating better learning, especially for students from disadvantaged backgrounds. MathLab aims to make a positive and lasting impact on the educational environment by making mathematics more accessible to all students by providing accessible, interactive, and personalized education.

REFERENCES

[1] S. Kumar, A. S. V. Prasad, "Automated Generation of Animated Videos for Question-Based Explanations", The International Journal of Innovative Engineering and Research (TIJER), TIJER, vol. 1, no. 1, pp. 1-8, 2023.
[2] K. R. Gupta, S. G. Sharma, "Real-Time Video to Text Summarization using Neural Network", International Research Journal of Engineering and Technology (IRJET), IRJET, vol. 7, no. 12, pp. 3435-3442, 2023.
[3] M. Kumar, J. L. J. Patil, "Intelligent Video Editing: Incorporating Modern Talking Face Generation", Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP), ICVGIP, vol. 16, no. 8, pp. 12-22, 2021.
[4] H. B. Shukla, R. K. Rao, "Image and Video Captioning Using Deep Learning", International Journal of Creative Research Thoughts (IJCRT), IJCRT, vol. 10, no. 3, pp. 101-108, 2023.
[5] R. K. Joshi, P. S. Dhamale, "Generative AI (Gen AI) Based Video Generation for Classical Dance", International Journal of Creative Research Thoughts (IJCRT), IJCRT, vol. 9, no. 1, pp. 22-29, 2023.
[6] M. K. Nivangune, S. B. Ghawate, H. B. Kasar, "Air-Writing Recognition System", International Journal of Research and Analytical Reviews (IJRAR), IJRAR, vol. 10, no. 4, pp. 273-278, 2023.
[7] A. K. R., N. H. R., P. J., "Air Writing Detection and Recognition", International Journal of Advanced Research in Science, Communication and Technology (IJARSCT), IJARSCT, vol. 2, no. 3, pp. 15-20, 2022.
[8] K. Navya, M. Sowmyah, S. A. Amreen, "A Machine Learning Approach for Air Writing Recognition", International Journal for Research in Applied Science & Engineering Technology (IJRASET), IJRASET, vol. 11, no. 3, pp. 1234-1240, 2023.
[9] P. Kumar, V. S. Vaishnavi, "Virtual Board: Air Canvas using OpenCV and MediaPipe", Journal of the Instrument Society of India, ISI, vol. 53, no. 2, pp. 123-130, 2023.
[10] A. K. Arundhathy, A. Ashfiya, "Air-Drawn Character Recognition using AI and Pattern Recognition Methodology", International Journal of Novel Research and Development (IJNRD), IJNRD, vol. 8, no. 2, pp. 806-812, 2023.
[11] S. Sharma, R. Gupta, "Design and Implementation of Air-Writing Recognition", Journal of Emerging Technologies and Innovative Research (JETIR), JETIR, vol. 5, no. 8, pp. 976-982, 2018.
[12] S. M. Patil, A. R. Patil, "Air-Writing and Recognition System", International Journal of Creative Research Thoughts (IJCRT), IJCRT, vol. 10, no. 4, pp. 1234-1240, 2024.
[13] C. H. Patil, S. S. Apte, "An Air-Written Real-Time Multilingual Numeral String Recognition System", Journal of Computer Science, JCS, vol. 20, no. 8, pp. 1712-1722, 2024.
[14] A. Tripathi, A. K. Mondal, L. Kumar, "SCLAiR: Supervised Contrastive Learning for User and Device Independent Airwriting Recognition", arXiv preprint arXiv:2111.12938, arXiv, vol. 2021, no. 11, pp. 1-20, 2021.
[15] C. Amma, M. Georgi, "Airwriting: Hands-free Mobile Text Input by Spotting and Continuous Recognition of 3D-Space Handwriting with Inertial Sensors", Proceedings of the 16th International Symposium on Wearable Computers, ISWC, vol. 16, no. 1, pp. 52-59, 2012.
[16] M. Karan, G. Angadi, "Transforming and Reforming the Indian Education System with Artificial Intelligence", International Journal of Education and Development using Information and Communication Technology, IJEDICT, vol. 19, no. 1, pp. 45-60, 2023.
[17] S. Sharma, R. K. Singh, "Artificial Intelligence in Indian Higher Education Institutions: A Comprehensive Review", Journal of Data Science and Management, JDSM, vol. 5, no. 2, pp. 101-115, 2023.
[18] P. M. Ramteke, "The Impact of Artificial Intelligence on Academic Activities in Indian Universities", International Journal of Advanced Research in Computer Science, IJARCS, vol. 12, no. 4, pp. 89-95, 2023.
[19] R. Venugopal, V. Mamatha, "Impact of Artificial Intelligence (AI) on Teaching and Learning in India's Higher Education Sector", IOSR Journal of Research & Method in Education, IOSR, vol. 13, no. 5, pp. 1-6, 2023.
[20] N. Khan, D. Bhanushali, "Strengthening e-Education in India using Machine Learning", arXiv preprint arXiv:2105.15125, arXiv, vol. 2021, no. 5, pp. 1-12, 2021.
[21] S. Ojha, A. Narendra, "From Robots to Books: An Introduction to Smart Applications of AI in Education (AIEd)", arXiv preprint arXiv:2301.10026, arXiv, vol. 2023, no. 1, pp. 1-18, 2023.
[22] A. Gupta, P. Sharma, "Assessing the Challenges and Opportunities of Artificial Intelligence in Indian Education", International Journal of Global Advanced Research, IJGAR, vol. 2, no. 1, pp. 50-58, 2023.
[23] G. Melquiond, "Plotting in a Formally Verified Way", arXiv preprint arXiv:2108.03974, arXiv, vol. 2021, no. 8, pp. 1-13, 2021.
[24] P. Papaphilippou, "Revisiting 3D Cartesian Scatterplots with a Novel Plotting Framework and a Survey", arXiv preprint arXiv:2406.06146, arXiv, vol. 2024, no. 6, pp. 1-18, 2024.
[25] M. I. Hossain, S. Kobourov, "Research Topics Map: rtopmap", arXiv preprint arXiv:1706.04979, arXiv, vol. 2017, no. 6, pp. 1-16, 2017.
[26] S. Aghaei, A. Daeichian, "A Graphical-Based Method for Plotting Local Bifurcation Diagram", arXiv preprint arXiv:2105.10938, arXiv, vol. 2021, no. 5, pp. 1-12, 2021.
[27] D. Drain, C. Wu, "Generating Bug-Fixes Using Pretrained Transformers", arXiv preprint arXiv:2104.07896, arXiv, vol. 2021, no. 4, pp. 1-14, 2021.
[28] Z. Fan, Y. Liu, "Automated Repair of Programs from Large Language Models", Proceedings of the 45th International Conference on Software Engineering, ICSE, vol. 45, no. 1, pp. 1125-1135, 2024.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY